SIGNAL AND IMAGE COMPRESSION USING DISCRETE WAVELET TRANSFORM

by

MOHD HAFIZI BIN KAMARUDIN

FINAL PROJECT REPORT

Submitted to the Electrical & Electronics Engineering Programme
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

SIGNAL AND IMAGE COMPRESSION USING DISCRETE WAVELET TRANSFORM

by

Mohd Hafizi Bin Kamarudin

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

____

(Mr. Samsul Ariffin Bin Abdul Ariffin)
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

December 2010

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

____

(Mohd Hafizi Bin Kamarudin)

ABSTRACT

Wavelet and Fourier transform are the common methods used in signal and image compression. Wavelet transform (WT) are very powerful compared to Fourier transform (FT) because its ability to describe any type of signals both in time and frequency domain simultaneously while for FT, it describes a signal from time domain to frequency domain. Because of that, the performance of FT is outperformed by the impressive ability of WT for most type of signals (stationary or non-stationary). Wavelets transform are able to describe any type of signals both in time and frequency domain. So the analysis can be done on signal at various scale and level which is important before we can do more analysis to the signal (stationary or non-stationary). The current standard for image processing in is JPEG2000 which is based on biorthogonal wavelets. Before that JPEG used Discrete Cosine Transform (DCT). This project will discuss the use of Fast Fourier Transform (FFT) and Fast Wavelet Transform (FWT) for signal and image compression. Some numerical experiments were done by using various types of signals/images and various wavelet filters such as Haar (2 filters) and Daubechies (up to 10 filters). Signal compression can be achieved by selecting the threshold value in order to cut-off the data when the high frequency components are filtered out. Based on Mean Squared Error (RMSE), Mean Squared Error (MSE) and Compression Ratio (CR), it is found that both methods are capable to compress the signal where Haar is the best for “Block” signal, D8 gives best result for “Mishmash” signal and FFT is the best for “Heavy Sine”. The plots of RMSE versus level of compression were shown where we can see clearly the trend of compression for all wavelet filters. All the numerical results were done by using Matlab programming.

For image compression, some numerical experiment will be doing by using various type of images (eg: Lena, Barbara, Mandrill and etc.) and various wavelet filters such as Haar (2 filters) and Daubechies (up to 20 filters). Image compression can be achieved by selecting the threshold value in order to cut-off the data when we filtered out the high frequency component. The analysis will be carried out in terms of Mean Squared Error (MSE), Root Mean Squared Error (RMSE) Peak Signal to Noise Ratio (PSNR) and Compression Ratio (CR). All numerical results were done by using Matlab programming (Wavelet Toolbox). The result is expected that proper selection of wavelet on the basis of nature of images will improve the quality as well as compression ratio remarkably. From the result, D8 is the best for image “Lena” for compression ratio of 5:1, 10:1 and 20:1. For image “Peppers”, Haar is the best for compression ratio of 5:1 and 10:1 but not for 20:1 where D4 is slightly higher than Haar in terms of PSNR value. For image “House”, Haar performed well for 10:1 and 20:1 but not for 5:1 where D8 is better. Lastly, for image “Cameraman”, D8 performed the best for all compression ratios (5:1, 10:1 and 20:1). For application, the KLCI time series data, among wavelet families, D8 is the best and also better than Fast Fourier Transform (FFT) method. So, the results will provide a good reference for application developers to choose a good wavelet compression system for their application.

ACKNOWLEDGEMENT

First and foremost, praise to Allah S.W.T. who has helped and encouraged me to complete the Final Year Project (FYP) quite successfully. Without His permission, this FYP will not be a reality.

Secondly, I would like to take this opportunity to express my deepest gratitude to all parties who involved in this project either directly or indirectly such as UTP lecturers, technician, friends and my family. They have been supportive in helping me doing the project.

My next appreciation goes to my FYP supervisor, Mr. Samsul Ariffin Bin Abdul Karim from Science and Applied Mathematics Department who has guided and given me opportunity to carry out the project. From this project, I have learnt lot of new things especially in signal and image processing area which is completely new for a beginner like me. Mr. Samsul has given me lot of support and knowledge that related to signal and image processing area especially about Wavelet which is the key of my project.

Last but not least, thanks also to others whose name was not mentioned in this page but has in one way or another contribution to the accomplishment of this project. Hopefully, this project will benefit anyone who has any interest in the field of signal and image processing.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background of Study

The technology nowadays has become more interesting as it comes to signal and image processing area. This led to the demand of better quality of signal with minimum size, lowest data rate and as well as the reliability. The main thing here is the process of data compression. Data compression is the process of reduction in size of data in order to save transmission time and space [1]. If the comparison was made between compressed and uncompressed data, we will notice that the data rate of compressed data is higher than the uncompressed data. For example, transmission and storage of uncompressed data would be extremely costly and impractical if we are dealing with huge or bundle of data. Consider a huge amount of data (eg: 1 Gb) that need to be transmitted, without compressing them first, the transmission time might be a couple of hours and to store those data, we need at least 1 Gb. If we compressed first, the transmission time will be reduced and storage space certainly would be less than 1 Gb.

In signal and image processing, wavelets have become one of the most needed tools. For example, signal and image compression, finance, computer vision and etc. One of the main reasons of successful wavelet method in signal or image compression is because it is a time and frequency based. The ability to ”zoom in” on any image or video data is another factor why wavelets are very popular tools. Furthermore, the use of fast algorithm which is Mallat algorithm to evaluate the wavelet transform with reduced number of arithmetic manipulation is very applicable and reliable. In order to compress data (signal or image), we have several choices like Fourier transform,Wavelet transform, Discrete Cosine

Transform (DCT-which is being used in JPEG and MPEG), Gabor transform and Wash-Hadamard transform. This paper will compare between Fast Fourier Transform (FFT) and Discrete Wavelet Transform (DWT).

Theoretically, not all type of signals can best be compressed using Wavelets in terms of error calculation. For a certain type of signal (such as sin or cosine), Fourier is more powerful than Wavelets because of the persistency of elimination of high frequency. Wavelets are powerful for certain type of signal due to huge percentage of zeroes but with high energy. In this paper, three types of signal were considered to prove each method of compression is the best for different type of signal. Then, all comparisons were made in terms of mean square error (MSE), root mean square error (RMSE) and compression ratio (CR).

Table 1: Multimedia Data [1]

Multimedia Data	**Size/ Duration**	**Bits/pixel or Bits/sample**	**Uncompressed size (B-bytes)**	**Transmission BW (b-bits)**	**Transmission Time**
Page of text	11” x 8.5”	Varying resolution	4-8 KB	32-64 Kb/page	1.1-2.2secs
Telephone Quality speech	10 seconds	8 bps	80 KB	64 Kb/Sec	22.2secs
Grayscale Image	512 x 512	8 bpp	262 KB	2.1 Mb/image	1min 13secs
Color Image	512 x 512	24 bpp	786 KB	6.29 Mb/image	3min 39secs
Medical Image	2048 x 2048	12 bpp	5.16 MB	41.3 Mb/image	23min 54secs

From the above information, the sufficient storage space, large transmission bandwidth and long transmission time are really needed. The current technology has an idea on how to handle those problems. The only way is by compressing the data first before storing or transmitting them. So, after transmitting the compressed data, it will be decompressed at the receiver. For clearer observation, consider this example. If we have 10 Gb of data that need to be transmitted or stored, we need at least 10 Gb of space and a couple of hours to send them out. If we did the compression first, let say for a compression ratio of 10:1, the space, bandwidth and transmission time requirements can be reduced by factor of 10 but the quality of the data still reliable. There are many methods of compression such as Fourier Transform (FT), Wash-Hadamard, Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT). This project will consider two methods which are Fourier Transform and Discrete Wavelet Transform. At the end, a few types of signals and images will be decomposed and compressed using those two methods and numerical comparison will be made to determine which method is more acceptable.

Speech coding has been and still is a major issue in the area of digital speech processing. Speech coding is the act of transforming the speech signal at hand, to a more compact form, which can then be transmitted with a considerably smaller memory. The motivation behind this is the fact that access to unlimited amount of bandwidth is not possible. Therefore, there is a need to code and compress speech signals. Speech compression is required in long-distance communication, high-quality speech storage, and message encryption. For example, in digital cellular technology many users need to share the same frequency bandwidth. Utilizing speech compression makes it possible for more users to share the available system. Another example where speech compression is needed is in digital voice storage. For a fixed amount of available memory, compression makes it possible to store longer messages [1].

Speech coding is a lossy type of coding, which means that the output signal does not exactly sound like the input. The input and the output signal could be distinguished to be different. Coding of audio however, is a different kind of problem than speech coding. Audio coding tries to code the audio in a perceptually lossless way. This means that even though the input and output signals are not mathematically equivalent, the sound at the output is the same as the input. This type of coding is used in applications for audio storage, broadcasting, and Internet streaming. Several techniques of speech coding such as

Linear Predictive Coding (LPC), Waveform Coding and Sub band Coding are existed. This project will attempt to study the wavelet compression technique on speech signals. The idea behind signal compression using wavelets is primarily linked to the relative scarceness of the wavelet domain representation for the signal. Wavelets concentrate speech information (energy and perception) into a few neighboring coefficients. Therefore as a result of taking the wavelet transform of a signal, many coefficients will either be zero or have negligible magnitudes. Data compression is then achieved by treating small valued coefficients as insignificant data and thus discarding them. The process of compressing a speech signal using wavelets involves a number of different stages, each of which are discussed below [1].

1.2 Problem Statement

Discrete Wavelet Transform has been widely used as a method of compressing a signal as well as for image. Although this technique gives a very good result but not many use this technique for analyzing the signal. Most of them apply this method for images since images look more interesting than the signal. In addition, because DWT has been very popular, it is very rare to see people do signal compression using Fast Fourier Transform method. So, the comparison of signal compression using DWT and FFT should be more precise to observe the differences between those two methods and thus will know which method gives the best result and how good the DWT method is, compare to FFT.

1.3 Objectives

The objectives of this project are;

i) To compare two methods of signal and image compression which are Discrete Wavelet Transform and Fast Fourier Transform

ii) To analyze the Discrete Wavelet Transform and Fast Fourier Transform techniques on various type signals and images.

1.4 Scope of Study

1.4.1 Fast Fourier Transform

It is very important to understand how Fourier analysis can be applied on signal compression. To be specific, Fast Fourier Transform (FFT) will be used for signal compression. The author needs to know how the decomposition of signal is done using FFT method. Then, the threshold level can be determined which is a part of compression ratio calculation. The Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) will be calculated to compare the result.

1.4.2 Discrete Wavelet Transform

The Discrete Wavelet Transform is studied to know the process of signal decomposition. It is different compare to Fast Fourier Transform because it uses coefficients such as *'details'* and *'approximation'*. Besides, to find the threshold level which gives optimum output, the author needs to use Graphical User Interface (GUI). For the moment, global threshold is used and the most accurate way is by using threshold by level but due to its complexity and for the time being, global threshold is preferred.

1.4.3 Signal (1-Dimension)

The various types of signals are chosen based on the complexity. So, the criteria where certain method is best for certain signal will be analyzed. To be fair, all the signals are fixed to 1024 points (length) and the comparison will be made using MSE, RMSE and compression ratio.

1.4.4 Image (2-Dimension)

For the image, few images will be considered based on the complexity. The size of all images is fixed to 1024 x 1024 or 512 x 512 (resolution). At the end, one case study will be considered such as compression on real data (e.g.: medical images).

CHAPTER 2

LITERATURE REVIEW

2.1 Signal Compression

In [1], the author discussed the use of WT (up to 10 filters) for speech compression. Speech compression is a process of converting human speech signals into efficient encoded representation that can be decoded back to produce a close approximation of the original signals. The input signal used is a 8 kHz 8-bit speech. Based on PSNR, SNR, NRMSE and compression ratio, they concluded D10 wavelet filter gives higher SNR and better speech quality with compression ratio up to 4.31 times and reduced the bit rate from 64 kbps to 13 kbps. In [2], the authors use wavelets to compress speech signal. They used spoken English speech to be analysed by D20 wavelet filter. The plot of SNR vs Compression Ratio (CR) was made and showed as CR goes higher, SNR gets lower. In [3], the author did the speech compression using Battle-Lemarie wavelet, Haar and Daubechies (up to 20 filters). The analysis was done by using voiced and unvoiced speech and the results shows Battle-Lemarie wavelet is the best while the other filters almost comparable except Haar. The numerical results were based on percentage of energy concentrated. While in [4], they analysed the effect of different compression schemes on speech signal. They used D4, D8, D10 and D20 and their input is Arabic speech signal (digit “0” and “8”). Based on SNR, PSNR and Normalized RMSE (NRMSE), they found that, by using smooth wavelets like D10, the percentage of truncated coefficients decreased and give better SNR. For unsmooth wavelet, it gives better compression ratio but with low SNR. Last but not least, in [5], the authors evaluated audio compression by using WT (up to 10 filters). Their main objective is to achieve transparent coding of audio and speech signal at the lowest possible data rate. Based on the numerical results, the found that D10 is the best wavelet filter with the lowest SNR and highest compression ratio (CR=1.88).

In this paper, it is quite different where the comparison will be made for signal compression using Fast Fourier Transform (FFT), Haar and Daubechies (up to 10 filters). Furthermore we do numerical comparison for all method and we also show the analysis starting level 1 for DWT. The analysis that we have done is more concrete and reliable. We also showed all the statistical measurement with the plots of RMSE versus level of compression for all wavelet filters length. All the results will be discussed more detail in result and discussion section.

Speech compression is a process of converting human speech signals into efficient encoded representations that can be decoded back to produce a close approximation of the original signals. A.M. Najih, et al. (2003) discovered for discrete wavelet transform, Db10 wavelet filter is the most suitable filter for speech compression compared to others Db and Haar which producing the highest SNR and better speech quality.

Audio compression has become one of the basic technologies of the multimedia age. In many applications, such as the design of multimedia workstations and high quality audio transmission and storage, the goal is to achieve transparent coding of audio and speech signals at the lowest possible data rates. However, if we can use less data, both transmission and storage become cheaper. Further reduction in bit rate is an attractive proposition in applications like remote broadcast lines, studio links, satellite transmission of high quality audio and voice over internet. In addition, other studies have shown that Db10 wavelet filter gives the highest SNR for audio compression (O.O. Khalifa, et al.,2008).

In other way, Discrete Fractional Fourier Transform gives better result in terms of percentage of non-zero coefficients compared to Discrete Fourier Transform. (Source: C. Vijaya and J.S. Bhat, "Signal Compression Using Discrete Fractional Fourier Transform and Set Partitioning in Hierarchical Tree",2004).

For image compression, (Javed Akhtar, 2006) had used SPIHT coding algorithm and wavelet decomposition. He used four types of wavelets which are Daubechies (up to 80 filters), Biorthogonal (up to 6.8), Coiflets (up to 5) and Symlets (up to 30) on a 256 x 256 image. The conclusion had been made based on Peak Signal to Noise Ratio (PSNR) and time taken for decomposition and reconstruction (seconds). From the result, db10, bior6.8, coif5 and sym25 give the highest PSNR which means they are the best for decomposition and SPIHT and most recommended for that kind of image.

In addition, EZW, SPIHT, SPECK, WDR and ASWDR algorithm had been compared by (R.Sudhakar, 2005) to find which algorithm is the best for decomposition of image. He used three images (lena, Barbara and cameraman) to do the observation. He used those algorithms which are the most recent which achieve some of the lowest errors per compression rate and highest perceptual quality. From his analysis, he compared those three algorithms on three different images and found that the optimum level that gives highest PSNR and better compression ratio is level 3 for. The best method is SPIHT which gives slightly higher PSNR value compared to the others.

Last but not least, (G.K.Kharate,2007) had compared compression performance of Daubechies, Biorthogonal, Coiflets and other wavelets for different frequency images. The images used by him are woman, wbarb and wmandril. From the numerical analysis, he concluded that db4 and bior2.4 performed significantly for those three images and db1 and bior1.1 performed significantly better for horizontal, vertical and diagonal line based images.

2.2 Fourier Transform

Traditionally, Fourier used to be one of the best methods for signal compression where complex signal are transformed to a much simpler one that is from time domain to frequency domain. Since Daubechies (1988) and Mallat (1989) have introduced compactly supported orthonomal wavelets with fast algorithm, the wavelets transform (WT) has been using in signal processing successfully [7]. For the comparison purpose, Fast Fourier Transform (FFT) had been applied to the three types of signals. Actually, FFT computes the Discrete Fourier Transform (DFT) and produces exactly the same result as evaluating the DFT definition directly but the only difference is that an FFT is much faster. The definition for DFT is as below:

$$X_k = \sum_{n=0}^{N-1} X_n e^{-i2\Pi k \frac{n}{N}} \quad \text{where } k=0,\dots\dots\dots,N-1 \tag{1}$$

FFT operates by decomposing an N point time domain signal into N time domain signals each composed of a single point. The next step is to calculate the N frequency spectra corresponding to these N time domain signals and then the N spectra are synthesized into a single frequency spectrum [8]. For example of FFT compression, consider a signal with a step function;

$$f(x) = \begin{cases} 1 & \text{if } -0.5 \le x < 0 \\ -1 & \text{if } 0 \le x < 0.5 \\ 0 & o.w \end{cases}$$

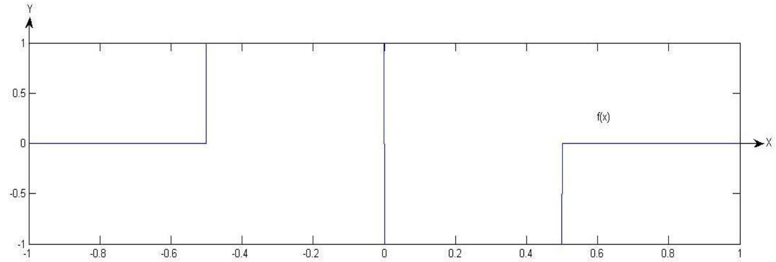


Figure 1(a): Original Signal

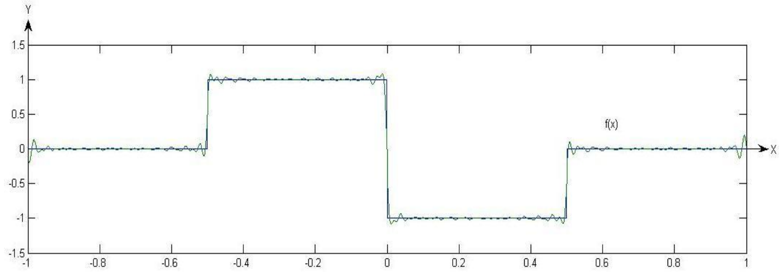


Figure 1(b): Compressed Signal

We can compress a signal by taking its FFT and then discard the small Fourier coefficients [9]. If we have 1024 sampled signal and we apply the FFT to transform the signal into frequency domain, we can decide the percentage of coefficients that we want to zero them. That is how we determine the compression ratio for signal compression using FFT.

The FFT is a complicated algorithm, and its details are usually left to those that specialize in such things. This section describes the general operation of the FFT, but skirts a key issue: the use of complex numbers [6].

In complex notation, the time and frequency domains each contain one signal made up of N complex points. Each of these complex points is composed of two numbers, the real part and the imaginary part. For example, when we talk about complex sample X[42], it refers to the combination of ReX[42] and ImX[42]. In other words, each complex variable holds two numbers. When two complex variables are multiplied, the four individual components must be combined to form the two components of the product. The following discussion on "How the FFT works" uses this jargon of complex notation. That is, the singular terms: signal, point, sample, and value, refer to the combination of the real part and the imaginary part [6].

The FFT operates by decomposing an N point time domain signal into N time domain signals each composed of a single point. The second step is to calculate the N frequency spectra corresponding to these N time domain signals. Lastly, the N spectra are synthesized into a single frequency spectrum [6].

Figure 1 shows an example of the time domain decomposition used in the FFT. In this example, a 16 point signal is decomposed through four separate stages. The first stage breaks the 16 point signal into two signals each consisting of 8 points. The second stage decomposes the data into four signals of 4 points. This pattern continues until there are N signals composed of a single point. An interlaced decomposition is used each time a signal is broken in two, that is, the signal is separated into its even and odd numbered samples. There are Log2N stages required in this decomposition, i.e., a 16 point signal (24) requires 4 stages, a 512 point signal (27) requires 7 stages, a 4096 point signal (212) requires 12 stages, etc [6].

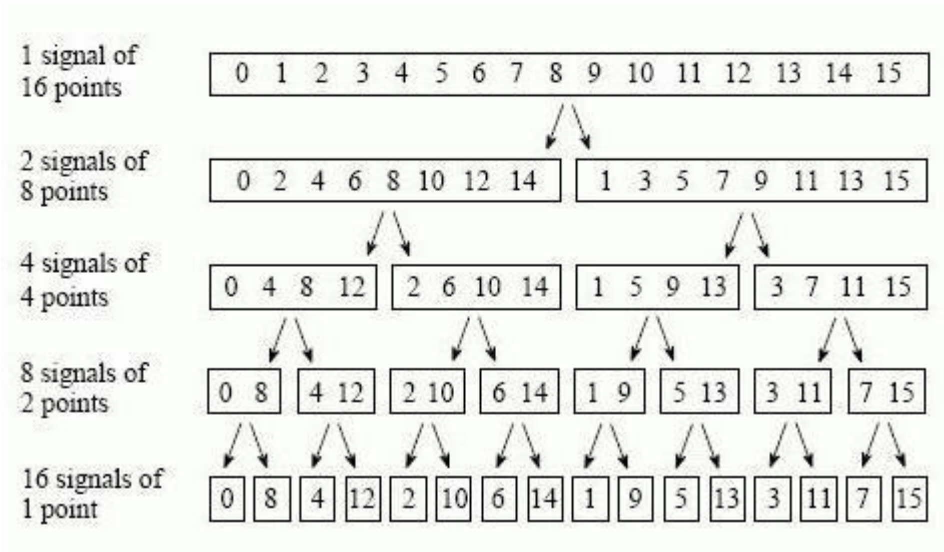


Figure 2: The FFT decomposition [6]

The next step in the FFT algorithm is to find the frequency spectra of the 1 point time domain signals. Nothing could be easier; the frequency spectrum of a 1 point signal is equal to itself. This means that nothing is required to do this step. Although there is no work involved, don't forget that each of the 1 point signals is now a frequency spectrum and not a time domain signal [6].

The last step in the FFT is to combine the N frequency spectra in the exact reverse order that the time domain decomposition took place. This is where the algorithm gets messy. Unfortunately, the bit reversal shortcut is not applicable, and has to go back one stage at a time. In the first stage, 16 frequency spectra (1 point each) are synthesized into 8 frequency spectra (2 points each). In the second stage, the 8 frequency spectra (2 points each) are synthesized into 4 frequency spectra (4 points each), and so on. The last stage results in the output of the FFT, a 16 point frequency spectrum [6].

2.3 Wavelet Transform

There are choices of wavelet basis function such as Haar, Daubechies, Meyer, Coiflet, Mexican Hat, biorthogonal wavelet and etc. Wavelet is defined from its scaling function (father wavelet) and wavelet function (mother wavelet)

([10],[11],& [12]). As mentioned previously, wavelet is compactly supported orthonomal where the function is

$$\phi(t) = 2^{j/2}(2^j t - k),\ j,k \in z \tag{2}$$

The wavelet series can be defined as below:

$$f(x) = \sum_k \alpha_k \varphi_{ok}(x) + \sum_{j=0}^{\infty}\sum_k \beta_{jk}\psi_{jk}(x) \tag{3}$$

Where α_k and β_k are coefficients defined by

$$\alpha_k = \int f(x)\varphi_{0k}(x)dx \tag{4}$$

$$\beta_k = \int f(x)\psi_{jk}(x)dx \tag{5}$$

Those two coefficients φ_{0k} and ψ_{jk} are called scaling function (father wavelet) and wavelet function (mother wavelet). In other term, φ_{0k} is also known as approximation while ψ_{jk} is called detail coefficients and $\{\varphi_{jk}, k\} \in Z$ is a basis for *Wj*. The equation in (3) is called a multiresolution expansion of f. The following expression is the wavelet expansion from (3).

$$\psi_{jk}(x) = 2^{\frac{j}{2}}\psi(2^j x - k),\ j,k \in Z \tag{6}$$

For the analysis and comparison, Haar and Daubechies were used. Haar is actually a part of Daubechies where it is also known as Daubechies 2 (2 filters). The wavelet and scaling functions for the Haar (D2) and Daubechies functions with order 2 up to 4 are shown below:

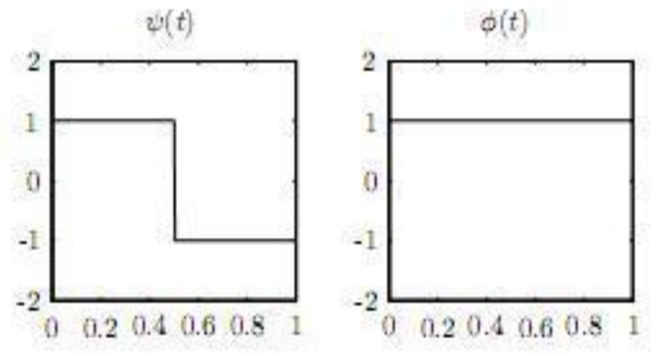


Figure 2a: Haar (D2)

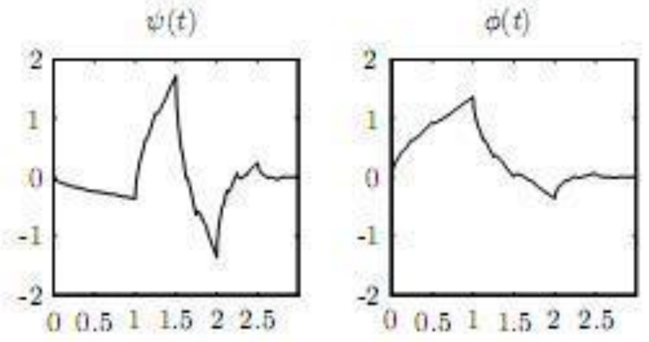


Figure 2b: Daubechies 4 (D4)

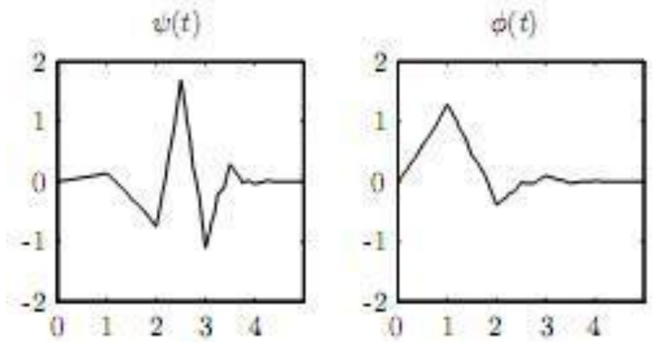


Figure 2c: Daubechies 6 (D6)

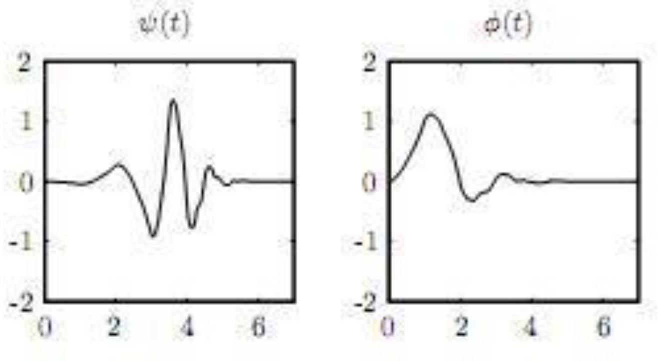


Figure 2d: Daubechies 8 (D8)

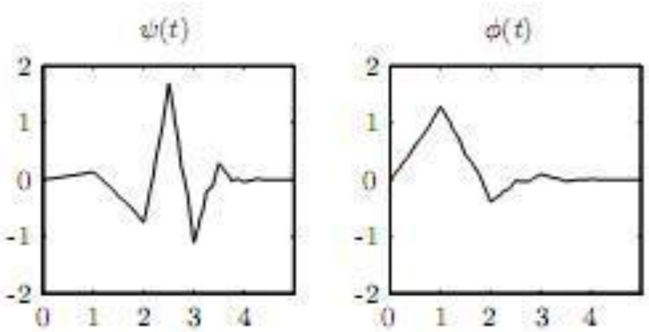


Figure 2e: Daubechies 10 (D10)

In this paper, we use Haar (2 filters) and Daubechies N (D N) where N is filter length such as D4, D6, D8 and etc. Some properties of of the Daubechies wavelets are asymmetric in particular for low values of N, orthogonal with compact support, the regularity inceases with order of N and the analysis id orthogonal. Theoretically, by looking at Haar scaling and wavelet function, we can say that it is very good for the signal such as step size or block signal but not really good for other types of signal such as sin, cosine and etc. Table 1 shows the summary of the filters that will be used in this project. Figure 3 shows the decomposition and reconstruction process of a signal. By having a signal, wavelet (low pass and high pass filter) will decompose or analyze that signal. It is a process of separating high (details) and low frequency (approximation) of coefficients (See Figure 4). Then, that signal can be compressed and to recover the signal the reconstruction process will take place. At the end, the signal produced is just similar to the original signal while maintaining the characteristics of the signal.

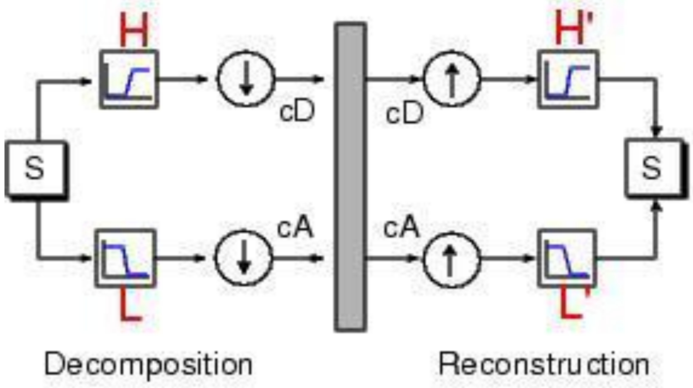


Figure 3: Signal Decomposition and Reconstruction

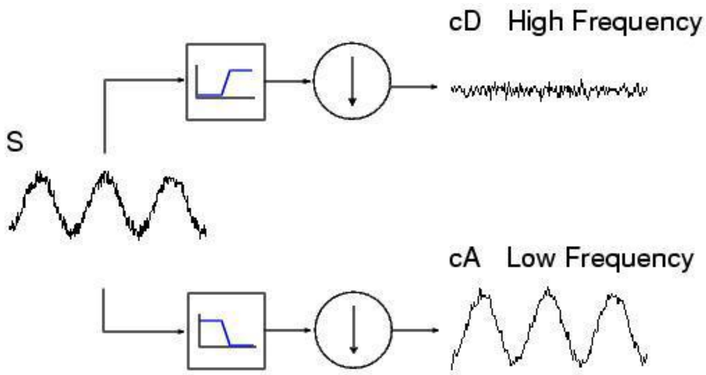


Figure 4: Details of Signal Decomposition

Table 2: Summary of used Wavelet filter

Filter's Name	**Short Form**	**Number of Filters**
Haar	D2	2
Daubechies 4	D4	4
Daubechies 6	D6	6
Daubechies 8	D8	8
Daubechies 10	D10	10

2.4 FFT versus DWT

2.5.1 Similarities

The fast Fourier transforms (FFT) and the discrete wavelet transform (DWT) are both linear operations that generate a data structure that contains $\log_2 n$ segments of various lengths. The mathematical properties of the matrices involved in the transforms are similar as well. The inverse transform matrix for both the FFT and the DWT is the transpose of the original. As a result, both transforms can be viewed as a rotation in function space to a different domain. For the FFT, this new domain contains basis functions that are sines and cosines. For the wavelet transform, this new domain contains more complicated basis functions called wavelets, mother wavelets, or analyzing wavelets. Both transforms have another similarity. The basic functions are localized in frequency, making mathematical tools such as power spectra [13].

2.5.2 Dissimilarities

The most interesting dissimilarity between these two kinds of transforms is that individual wavelet functions are *localized in space*. Fourier sine and cosine functions are not. This localization feature, along with wavelets' localization of frequency, makes many functions and operators using wavelets "sparse" when transformed into the wavelet domain. This sparseness, in turn, results in a number of useful applications such as data compression and removing noise from time series. One way to see the time-frequency resolution differences between the Fourier transform and the wavelet transform is to look at the basis function coverage of the time-frequency plane. Figure 10 shows a windowed Fourier transform, where the window is simply a square wave. Because a single window is used for all frequencies in the WFT, the resolution of the analysis is the same at all locations in the time-frequency plane [13].

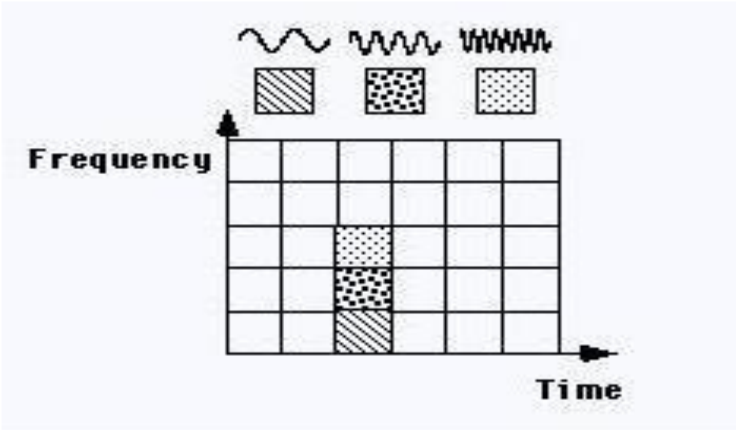


Figure 5: Fourier basis functions, time-frequency tiles, and coverage of the time-frequency plane [13]

An advantage of wavelet transforms is that the windows vary. In order to isolate signal discontinuities, one would like to have some very short basis functions. At the same time, in order to obtain detailed frequency analysis, one would like to have some very long basis functions. A way to achieve this is to have short high-frequency basis functions and long low-frequency ones. This happy medium is exactly what you get with wavelet transforms. Figure 11 shows the coverage in the time-frequency plane with one wavelet function, the Daubechies wavelet [13].

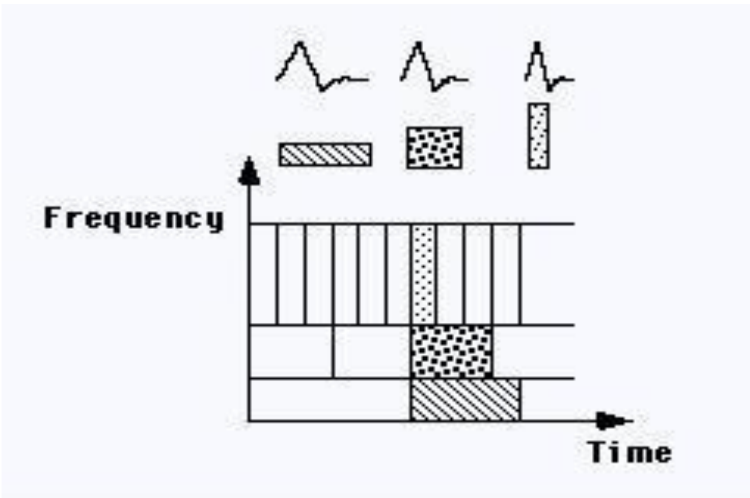


Figure 6: Daubechies wavelet basis functions, time-frequency tiles, and coverage of the time-frequency plane [13]

One thing to remember is that wavelet transforms do not have a single set of basic functions like the Fourier transform, which utilizes just the sine and cosine functions. Instead, wavelet transforms have an infinite set of possible basis functions. Thus wavelet analysis provides immediate access to information that can be obscured by other time-frequency methods such as Fourier analysis [13].

CHAPTER 3

METHODOLOGY

3.1 Procedure Identification

In Figure 7 on the next page show the flowchart of the project. Overall this project is divided into three parts. First part of the project is signal compression using Discrete Wavelet Transform (DWT) and Fast Fourier Transform (FFT). For DWT, two types of filter have been used which are Haar and Daubechies or from 2 filters(Haar) to 10 filters (D10). Using MATLAB software, both methods (FFT and DWT) have been applied to five types of signals which represent different complexity such as step or block function, sine or cosine and etc. Then, the comparison has been made by calculating compression ratio, MSE and RMSE.

For second part of project is image compression using Discrete Wavelet Transform (DWT) and Fast Fourier Transform (FFT). For DWT, default filter in Matlab 2007b has been used which is Embedded Zero Tree (EZW). For easier comparison with worldwide researches, common images have been used such as Lena, Cameraman, Peppers and etc. Then, the comparison will be made by calculating the errors (MSE and RMSE), compression ratio and Peak Signal-to-Noise Ratio (PSNR).

For the last part which is application, the real data has been used which are Kuala Lumpur Composite Index (KLCI) data which contains of 1075 coefficients. Using the data, DWT and FFT have been applied to find the best filter to compress it. Other application that will be used is either Electroencephalography (EEG) or Electrocardiogram (ECG). The result for this last part (EEG/ECG) is ongoing.

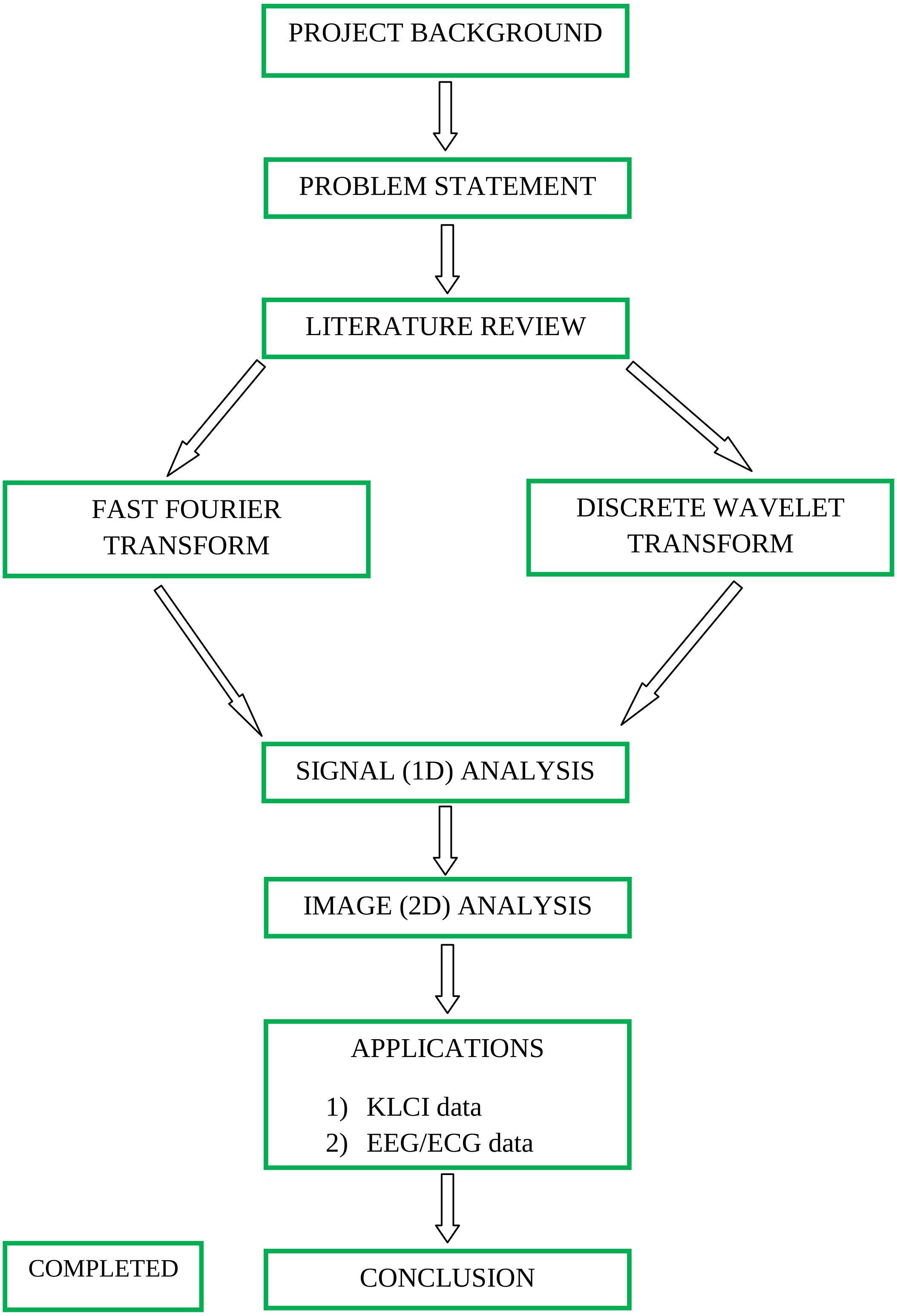


Figure 7: Flowchart of Project

3.2 Detailed Procedure

Throughout this project, there are some procedures to be followed. This is to ensure that the project can be accomplished within the given timeframe.

3.2.1 Software

The software will be used throughout this project. Wavelet Toolbox will be used fully for DWT analyzing. Figure 8 shows the interface of Wavelet Toolbox.

Figure 8: Interface of Wavelet Toolbox

3.2.2 Signal (1 Dimension)

Figure 9 below shows the flows for the first part of this project which is signal compression using DWT and FFT.

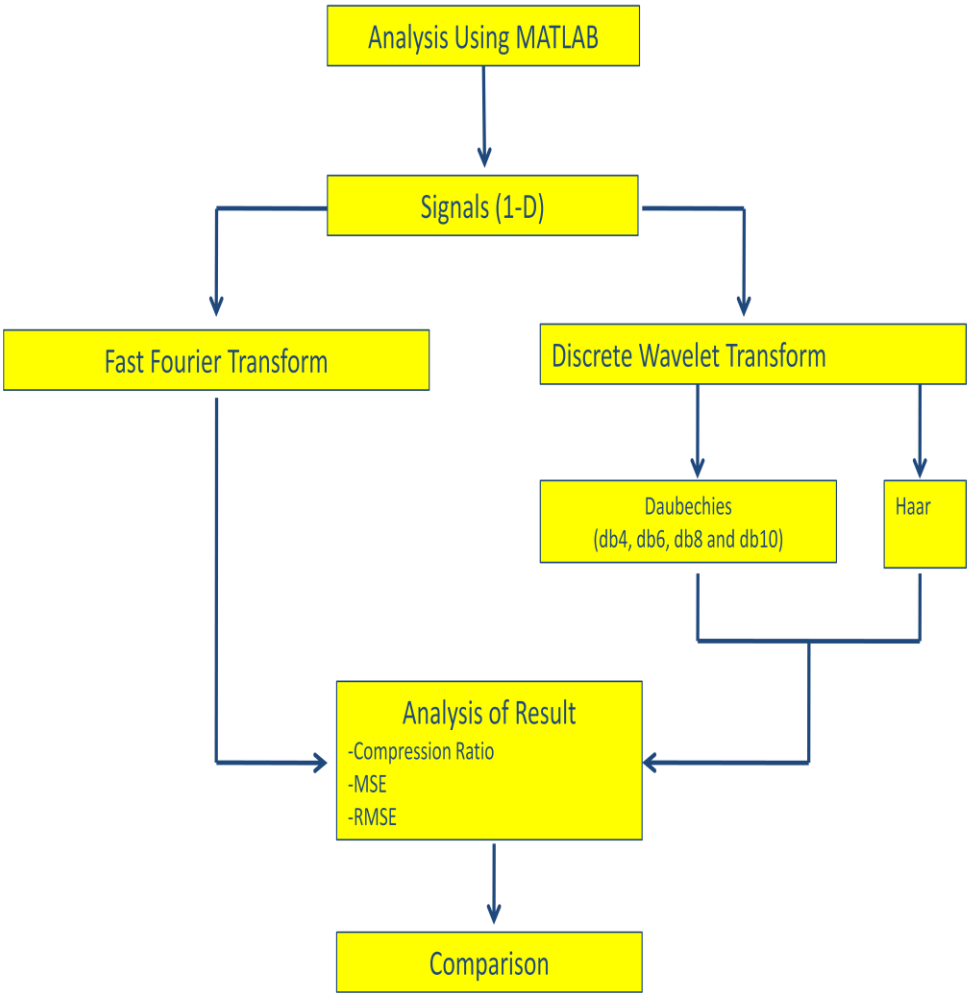
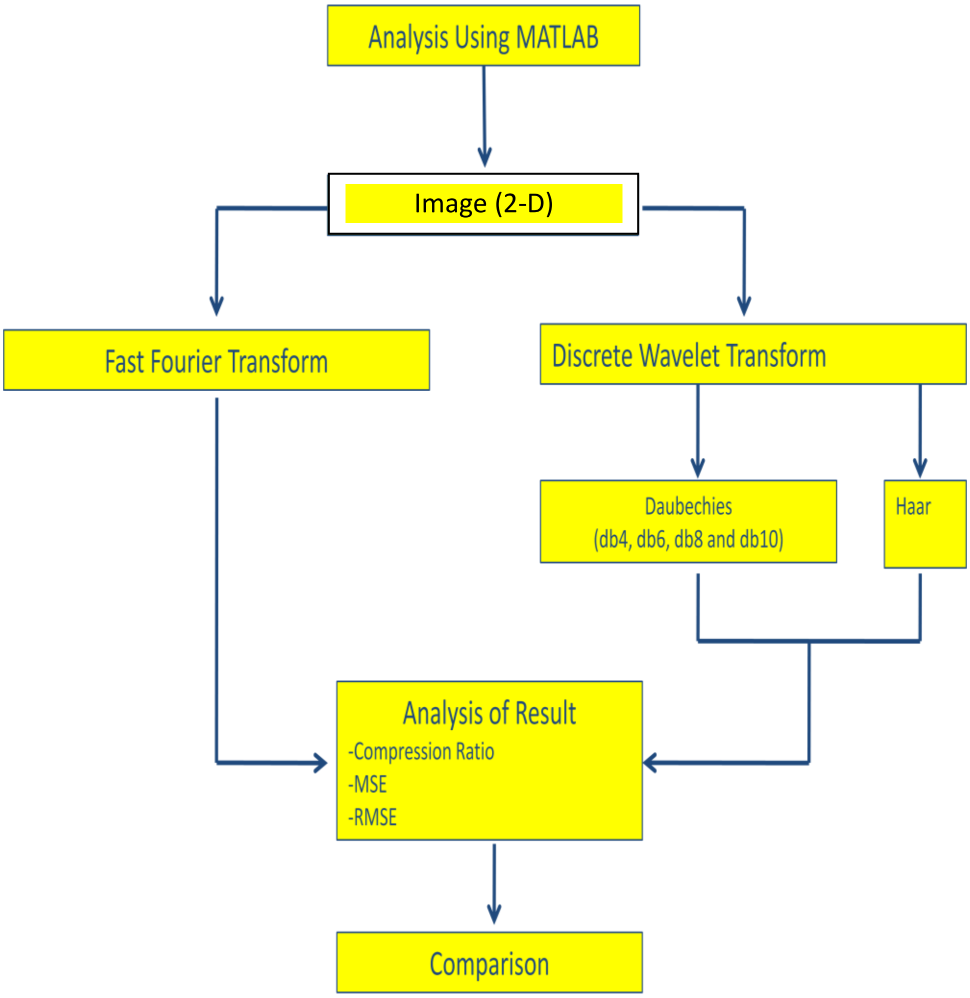


Figure 9: Flow for signal compression

3.2.3 Image (2 Dimension)



CHAPTER 4

RESULTS AND DISCUSSION

For the first part which is signal compression, three types of signals were used which are “Block”, “Heavy Sine” and “Mishmash” (see Figure 10). The reason why these three signals were chosen is to represent 3 common different types or categories of signals. For signal 1 which is “Block”, it is a step function where theoretically Haar filter will give the best result. The second signal which is “Heavy Sine”, it represents sine or cosine type of signal. Normally, for this type of signal, Fast Fourier Transform (FFT) will produce quite good result in terms of root mean squared error (RMSE). While for the third signal which is “Mishmash”, it belongs to neither block function nor sine or cosine type. So, the results will show which filter is the best especially among the Daubechies filters (D4 to D10).

Next, there are 2 methods that have been used to analyze these three signals which are FFT and Discrete Wavelet Transform (DWT). For FFT method, 10 different values of compression ratio (CR) (from 0 to 1) have been considered. The compression ratio actually represents the scale of zeros or percentage of coefficients that want to be eliminated. For example, a compression ratio of 0.8 is equal to 80% of zeros coefficients. For the DWT method, 5 types of filters have been used (Haar or D2, D4, D6, D8 and D10). For each filter, the analysis was done for 10 levels of decomposition. The results were compared in terms of RMSE, CR, Retained Energy (RE) and Number of Zeros (NOZ). For a fair comparison, the length or the number of coefficients/points of all the tested signals is the same which is 1024.

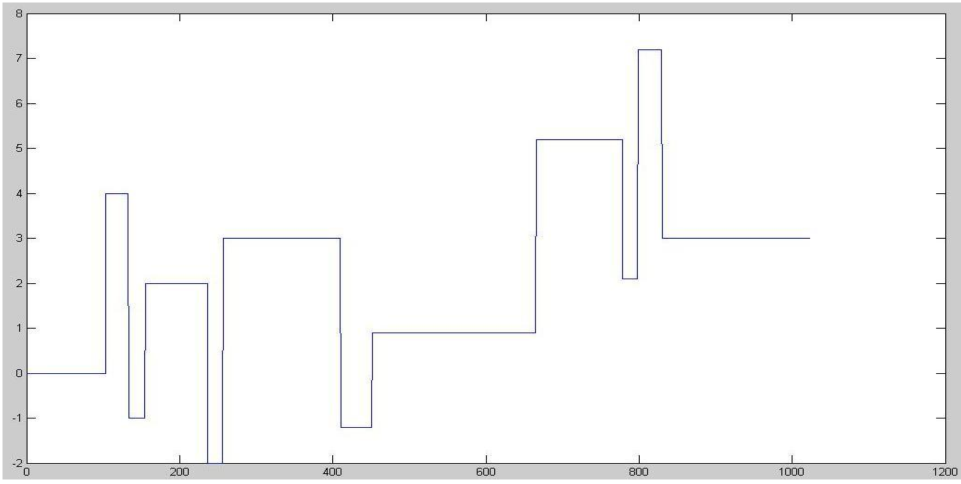


Figure 10a: Original Signal 1 (“Block”)

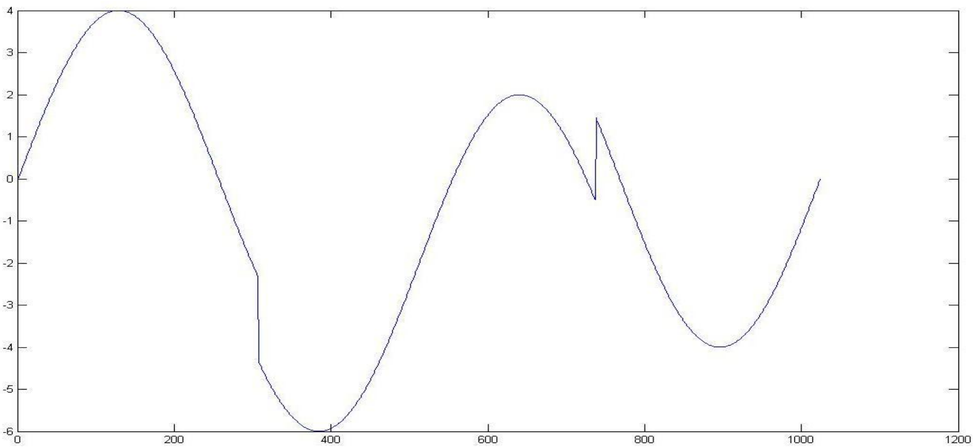


Figure 10b: Original Signal 2 (“Heavy Sine”)

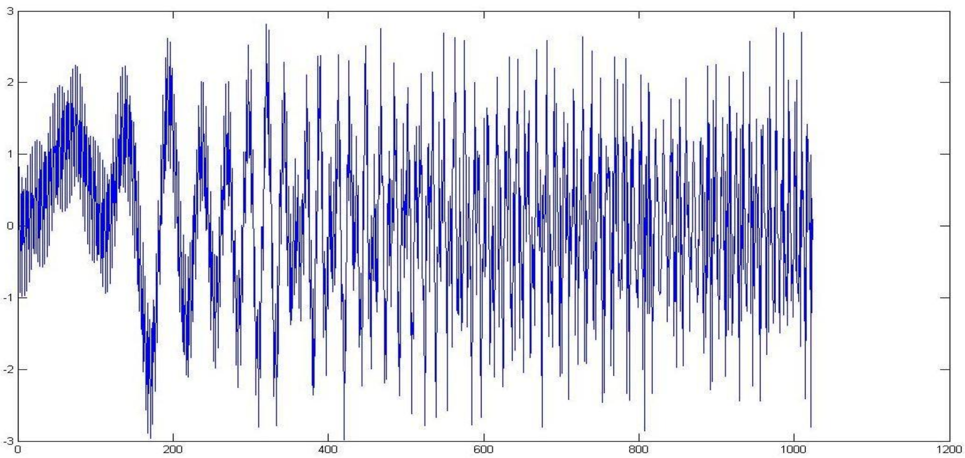


Figure 10c: Original Signal 3 (“Mishmash”)

4.1 Signal Compression Using Fast Fourier Transform (FFT)

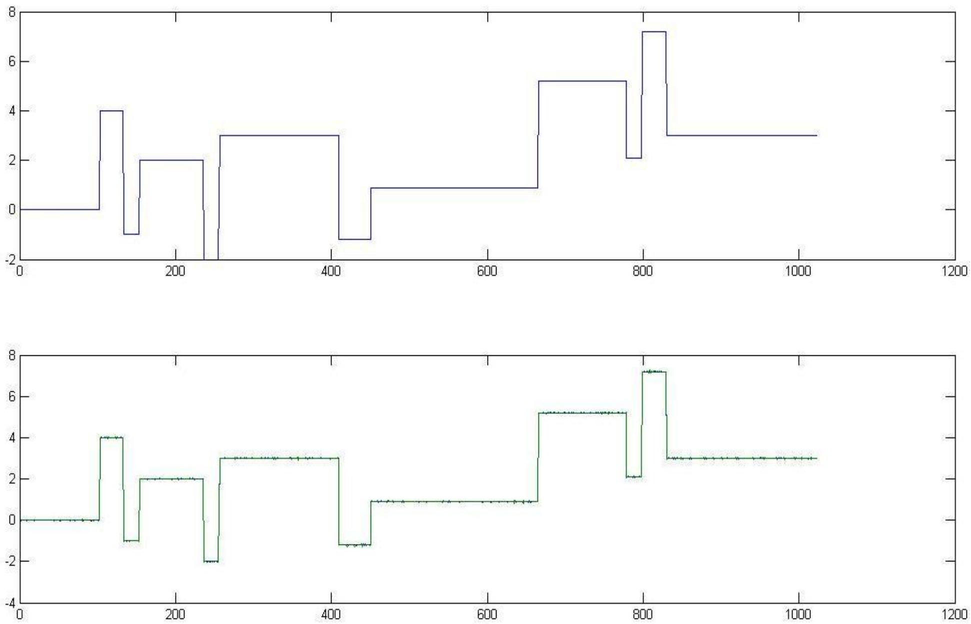


Figure 11: Compression Ratio 0.1 (upper-original and lower-compressed)

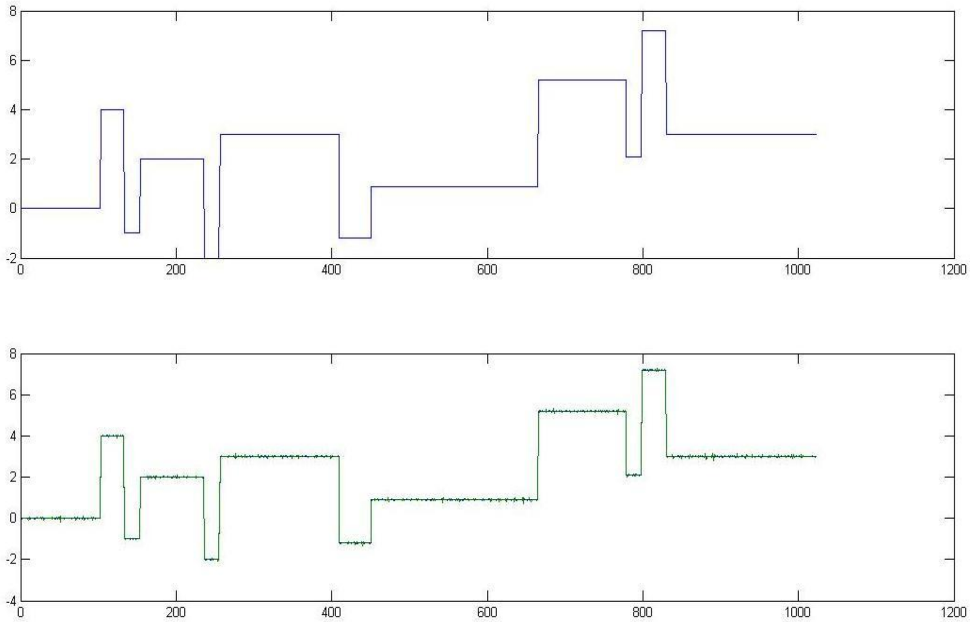


Figure 12: Compression Ratio 0.2 (upper-original and lower-compressed)

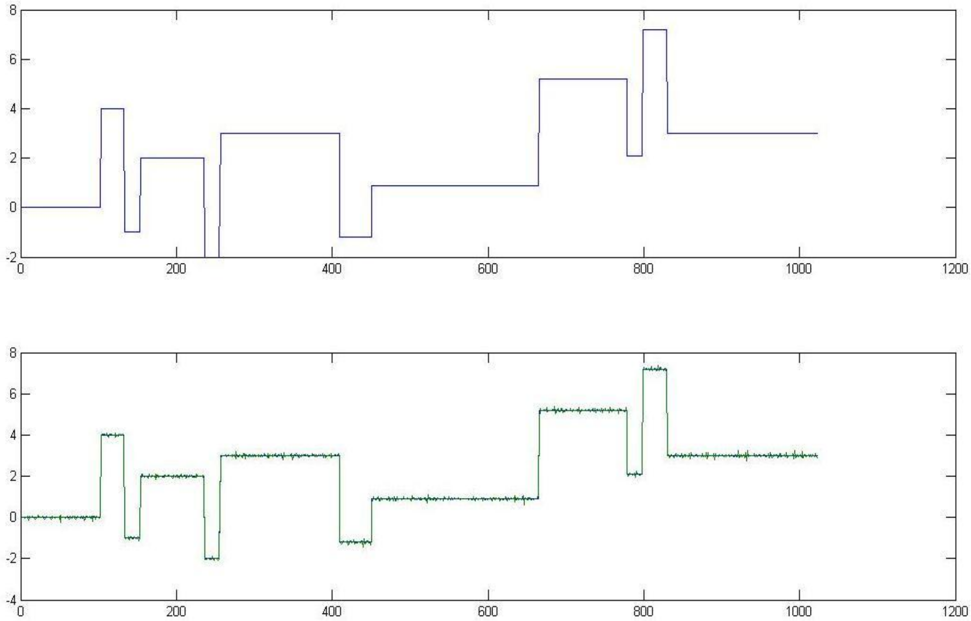


Figure 13: Compression Ratio 0.3 (upper-original and lower-compressed)

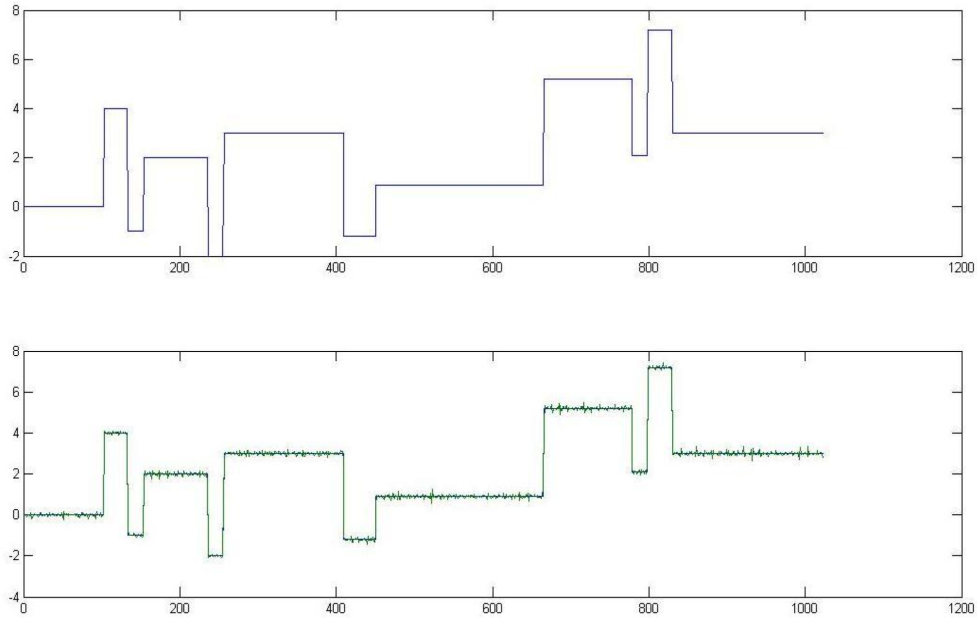


Figure 14: Compression Ratio 0.4 (upper-original and lower-compressed)

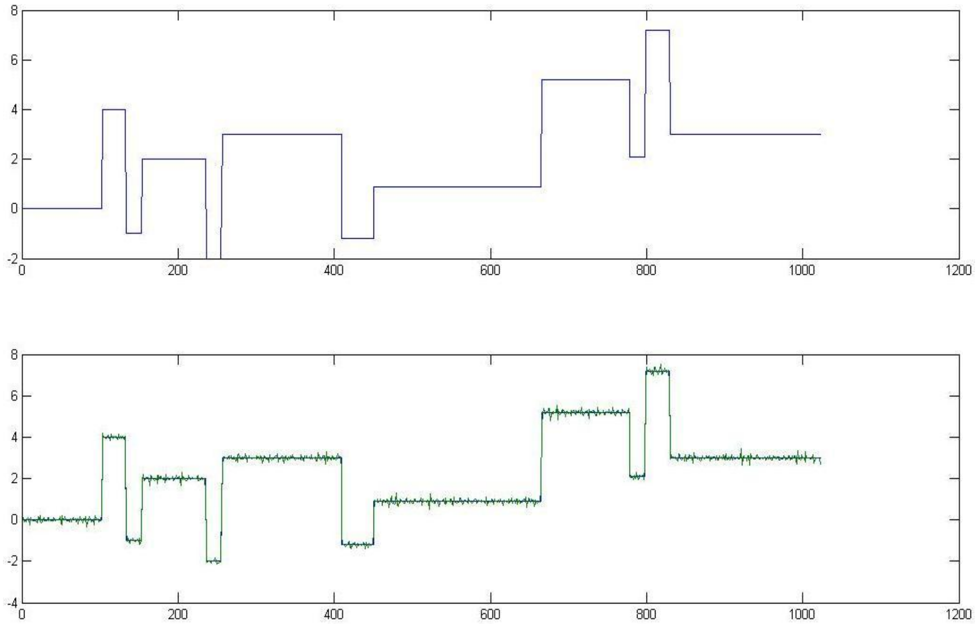


Figure 15: Compression Ratio 0.5 (upper-original and lower-compressed)

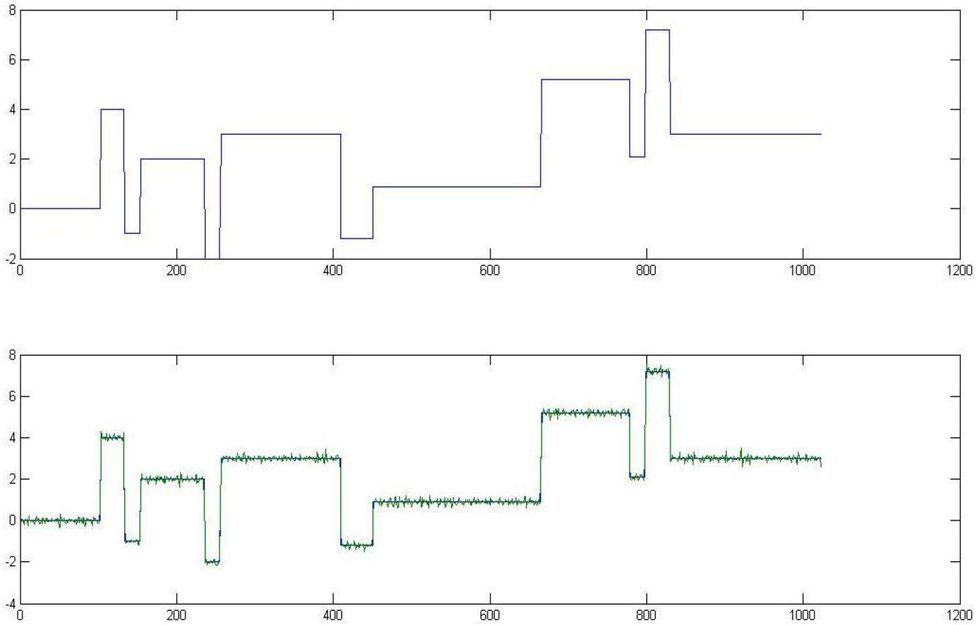


Figure 16: Compression Ratio 0.6 (upper-original and lower-compressed)

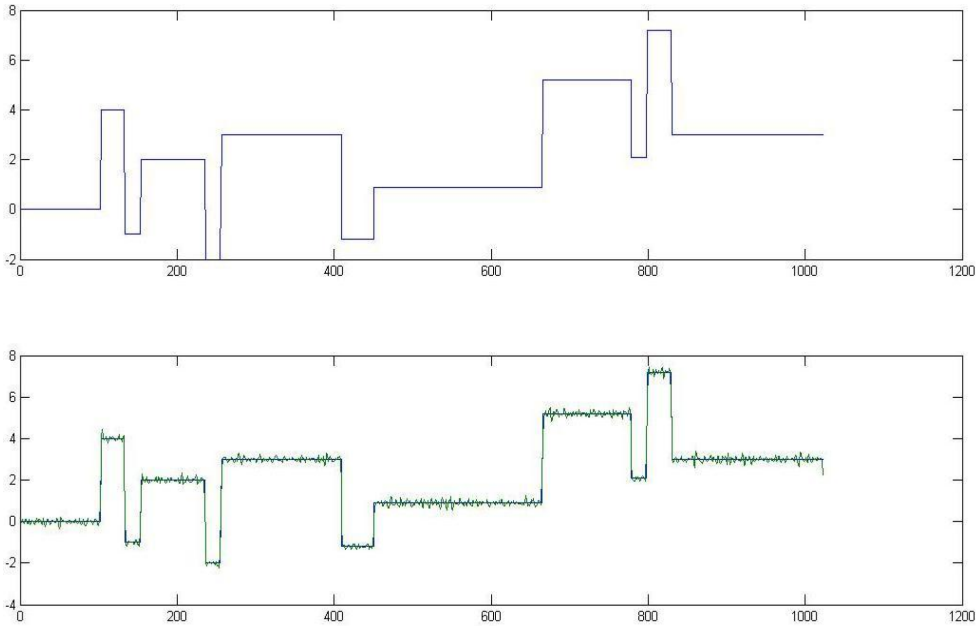


Figure 17: Compression Ratio 0.7 (upper-original and lower-compressed)

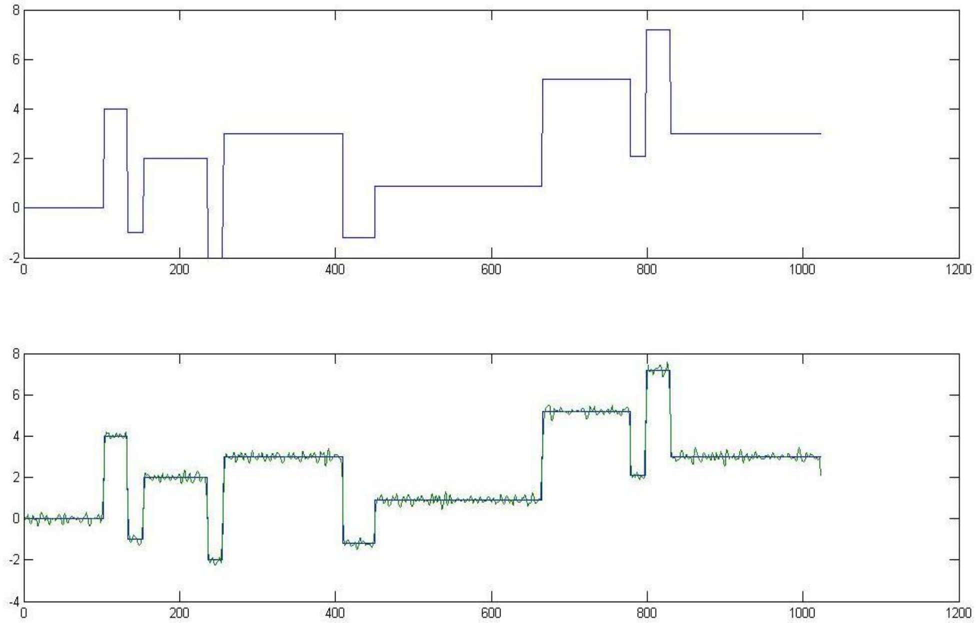


Figure 18: Compression Ratio 0.8 (upper-original and lower-compressed)

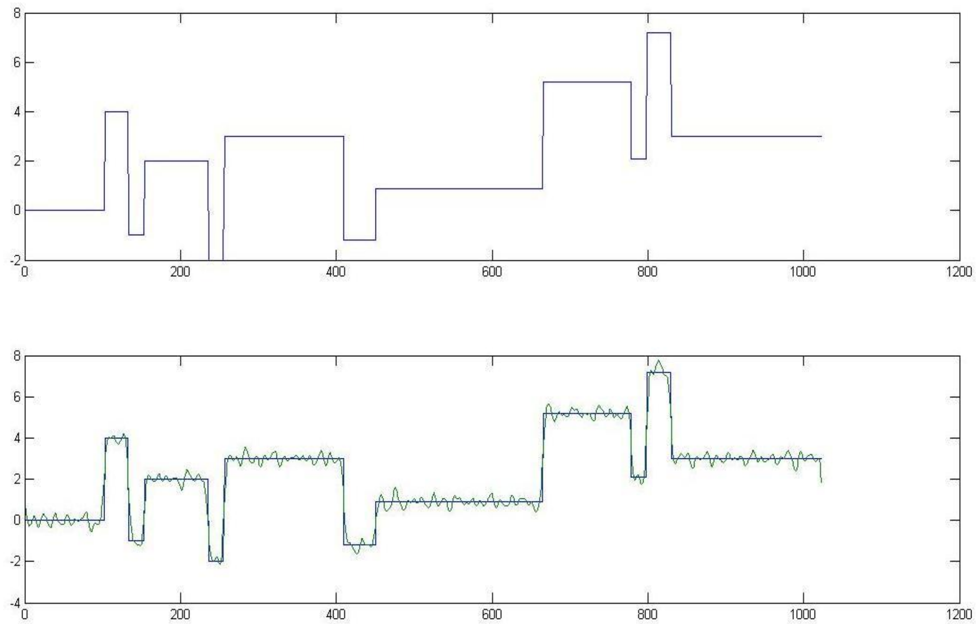


Figure 19: Compression Ratio 0.9 (upper-original and lower-compressed)

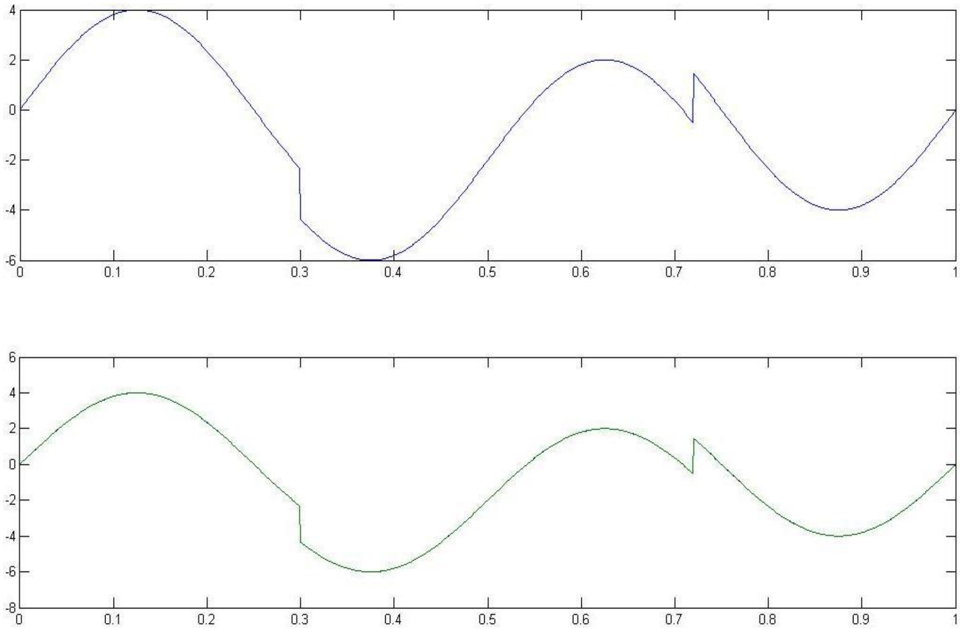


Figure 20: Compression Ratio 0.1 (upper-original and lower-compressed)

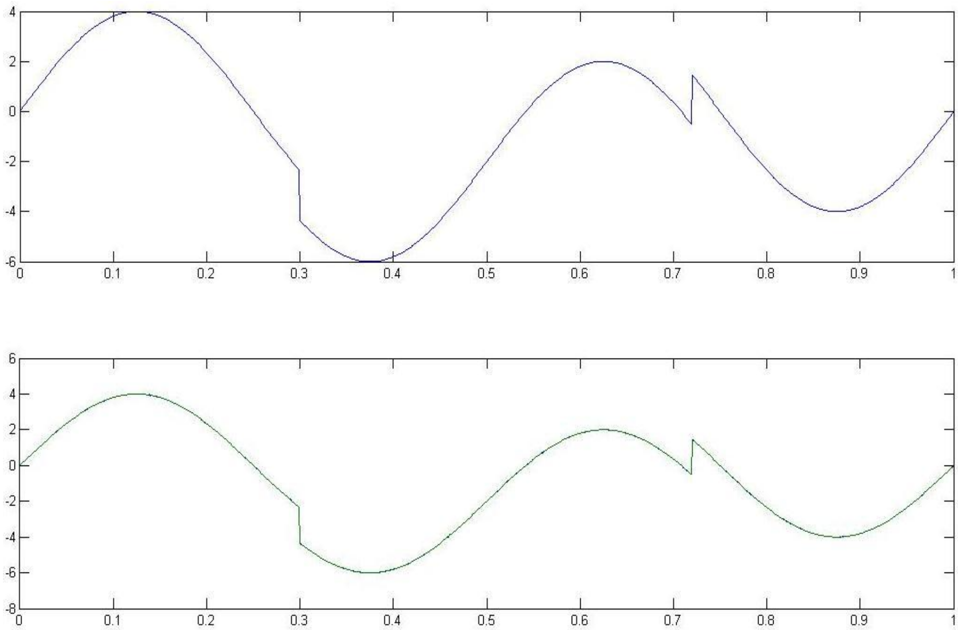


Figure 21: Compression Ratio 0.2 (upper-original and lower-compressed)

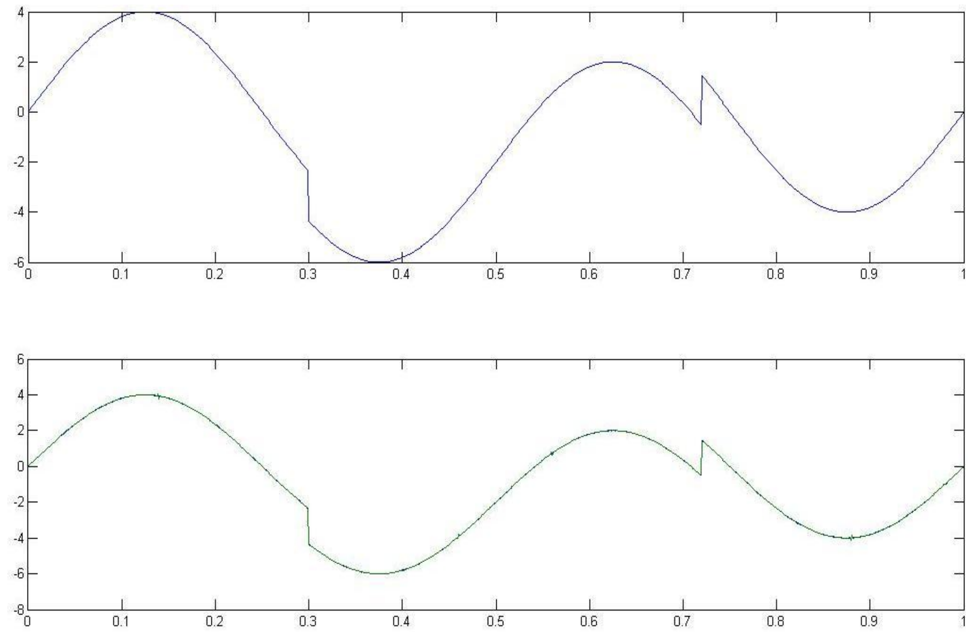


Figure 22: Compression Ratio 0.3 (upper-original and lower-compressed)

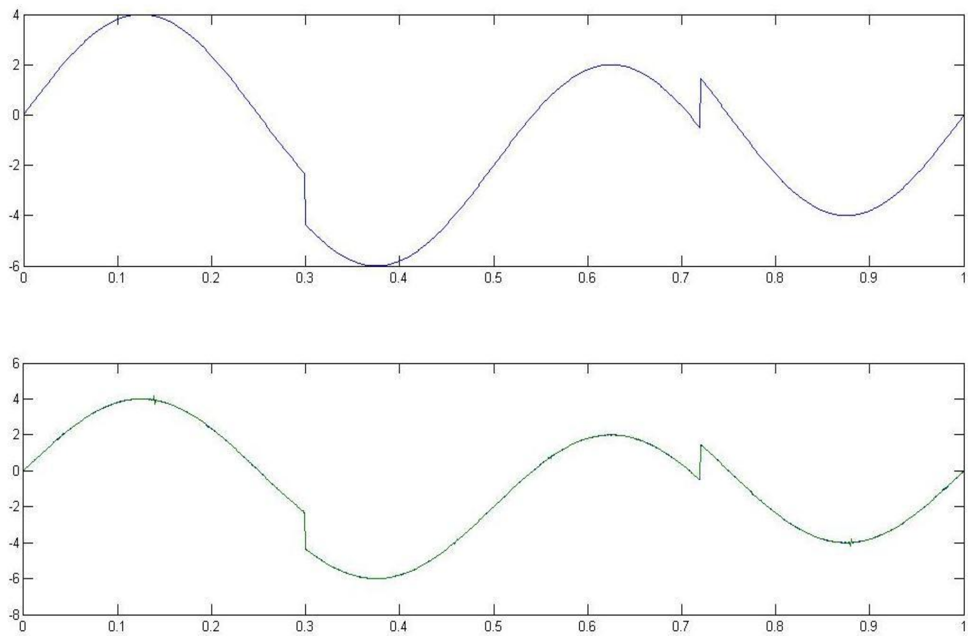


Figure 23: Compression Ratio 0.4 (upper-original and lower-compressed)

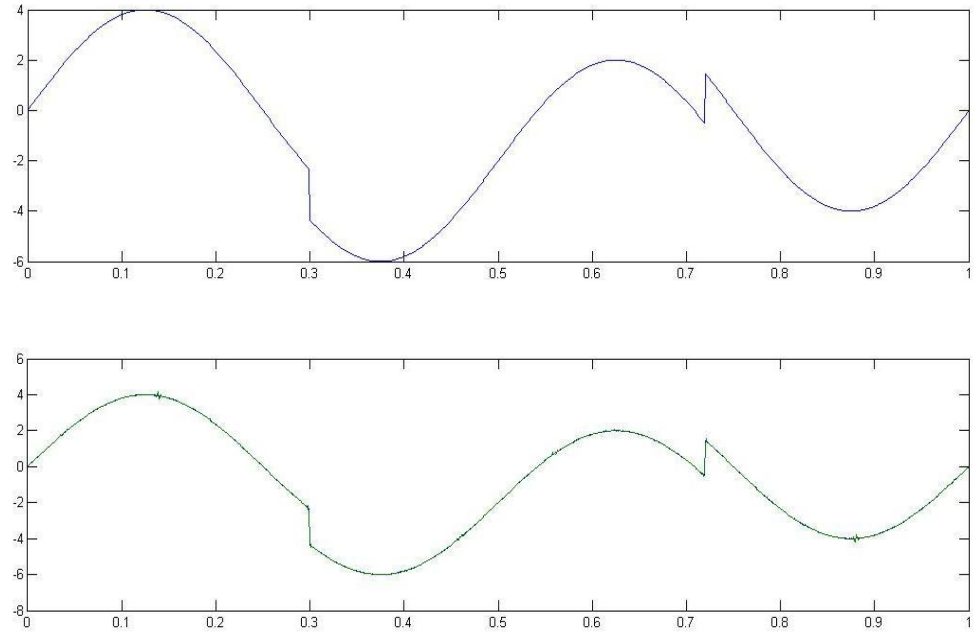


Figure 24: Compression Ratio 0.5 (upper-original and lower-compressed)

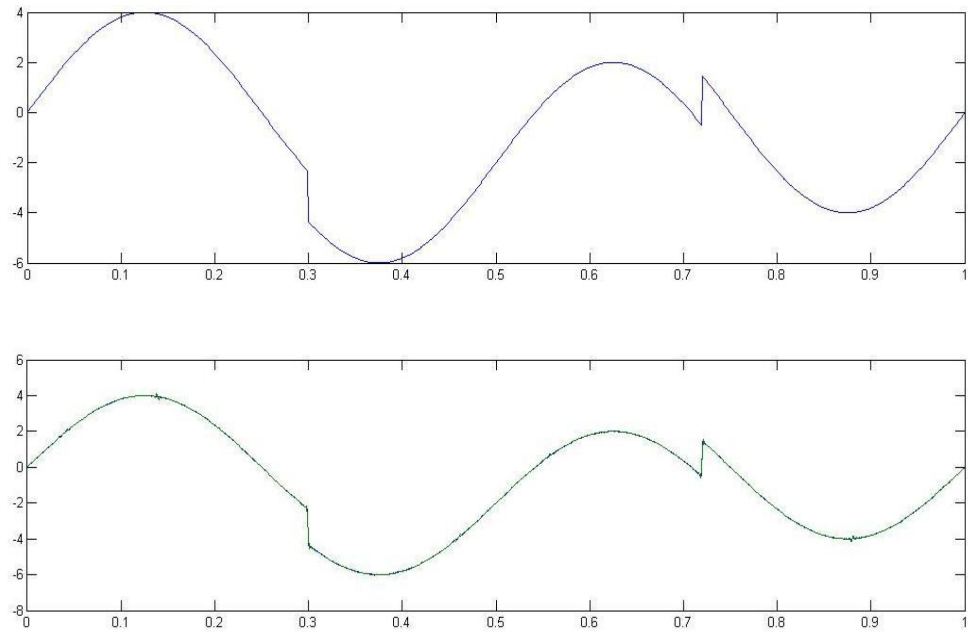


Figure 25: Compression Ratio 0.6 (upper-original and lower-compressed)

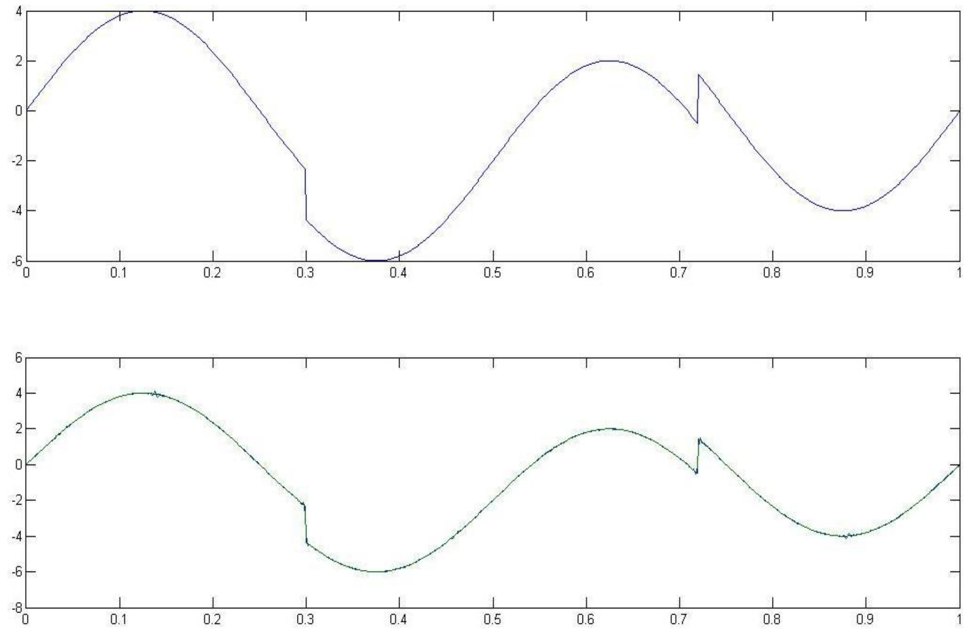


Figure 26: Compression Ratio 0.7 (upper-original and lower-compressed)

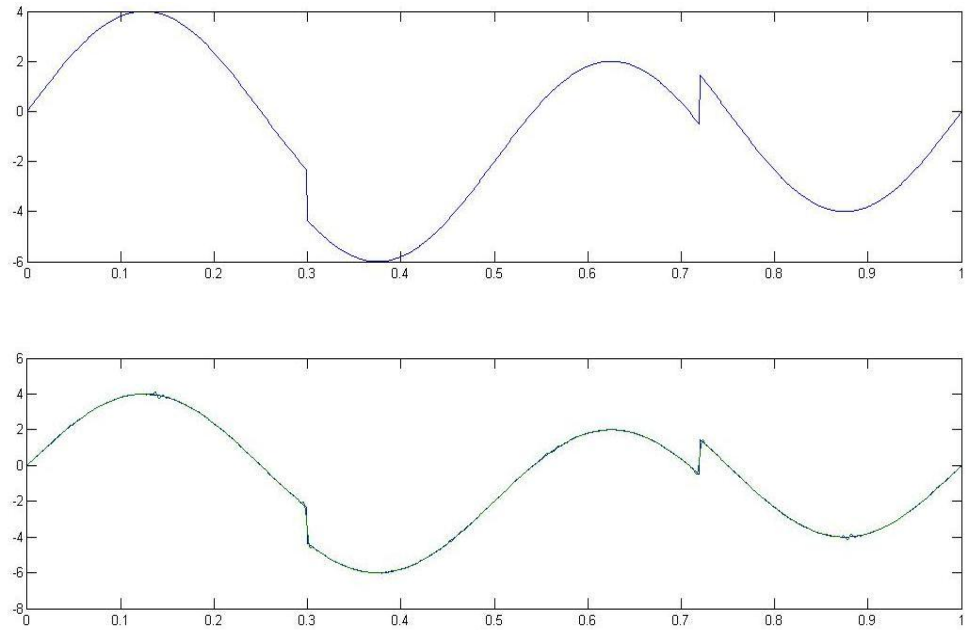


Figure 27: Compression Ratio 0.8 (upper-original and lower-compressed)

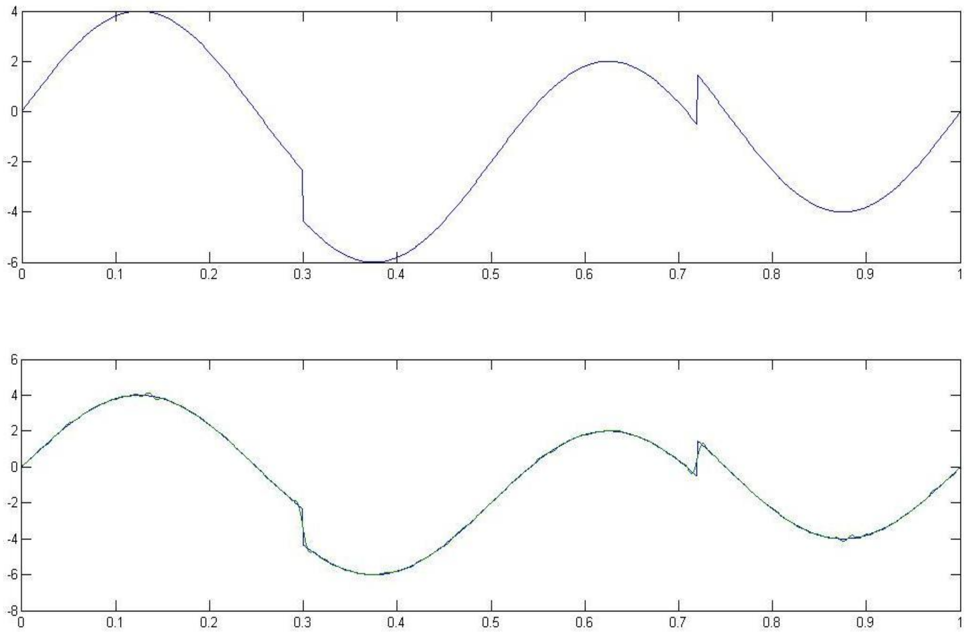


Figure 28: Compression Ratio 0.9 (upper-original and lower-compressed)

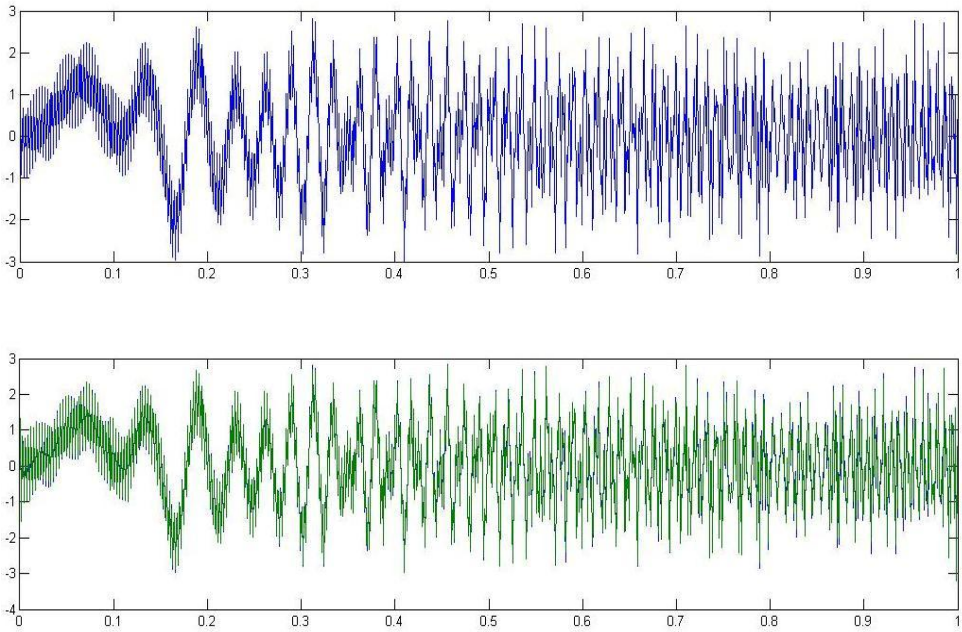


Figure 29: Compression Ratio 0.1 (upper-original and lower-compressed)

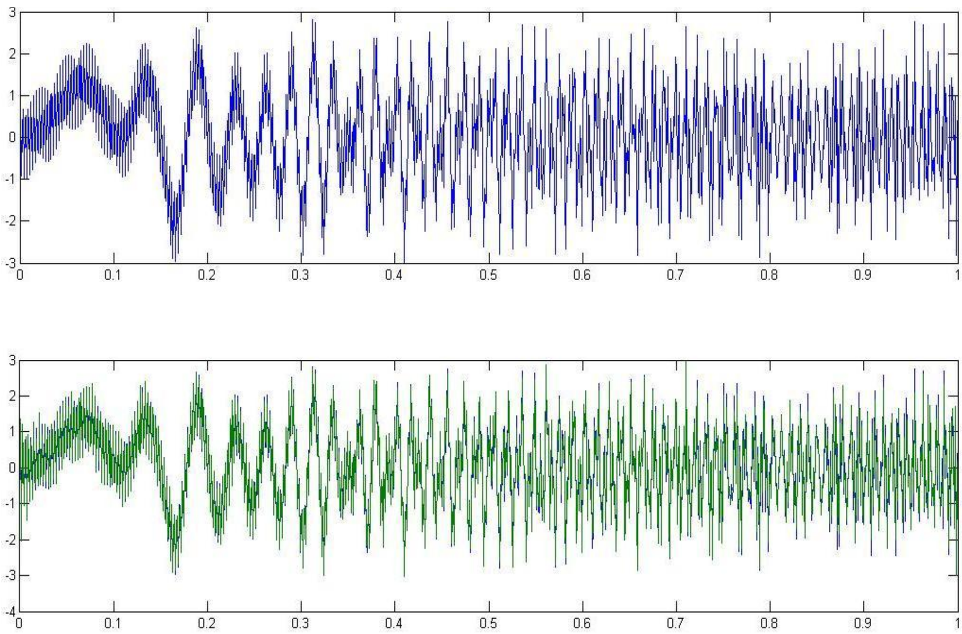


Figure 30: Compression Ratio 0.2 (upper-original and lower-compressed)

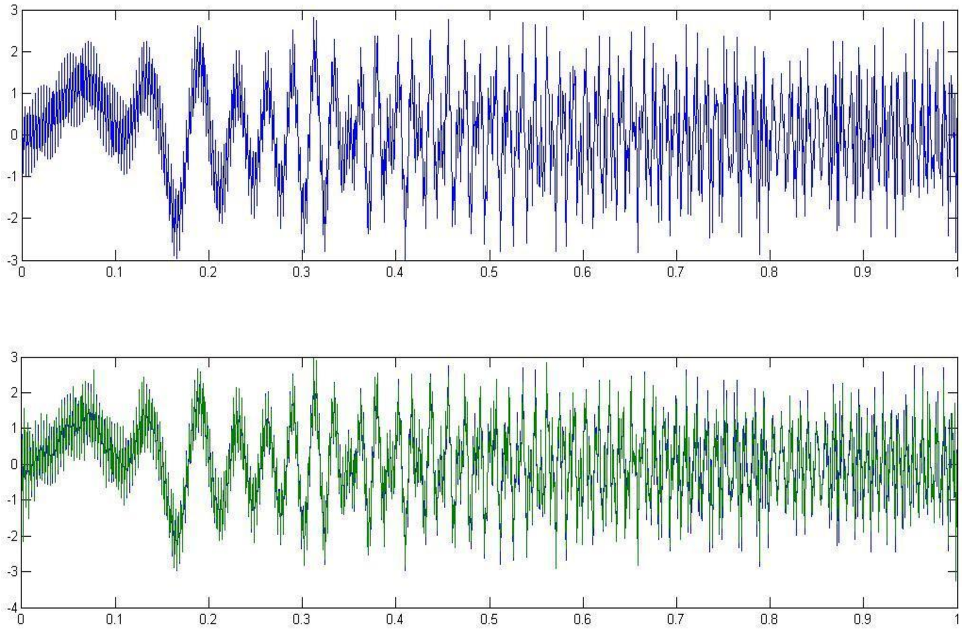


Figure 31: Compression Ratio 0.3 (upper-original and lower-compressed)

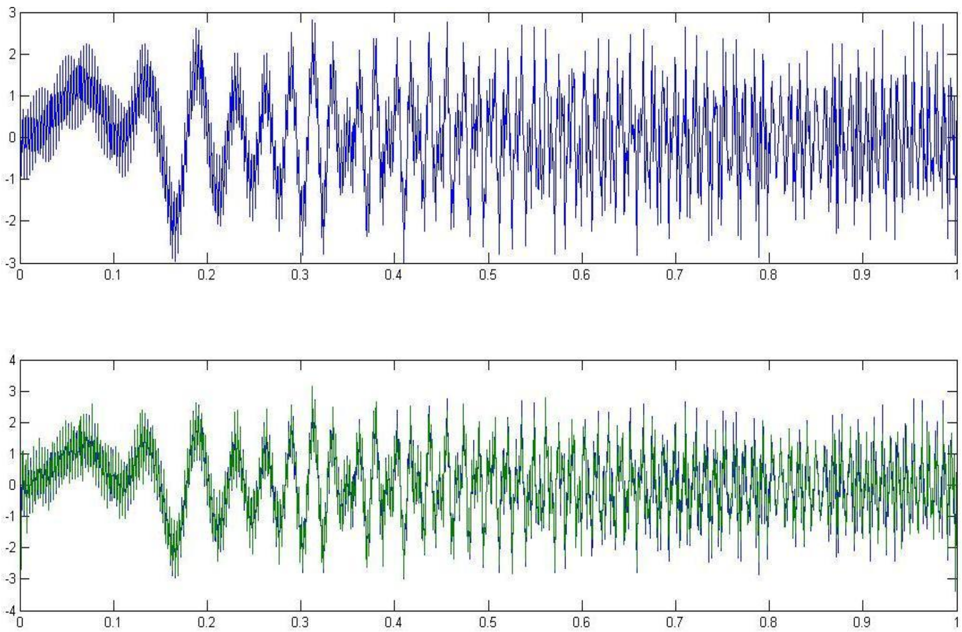


Figure 32: Compression Ratio 0.4 (upper-original and lower-compressed)

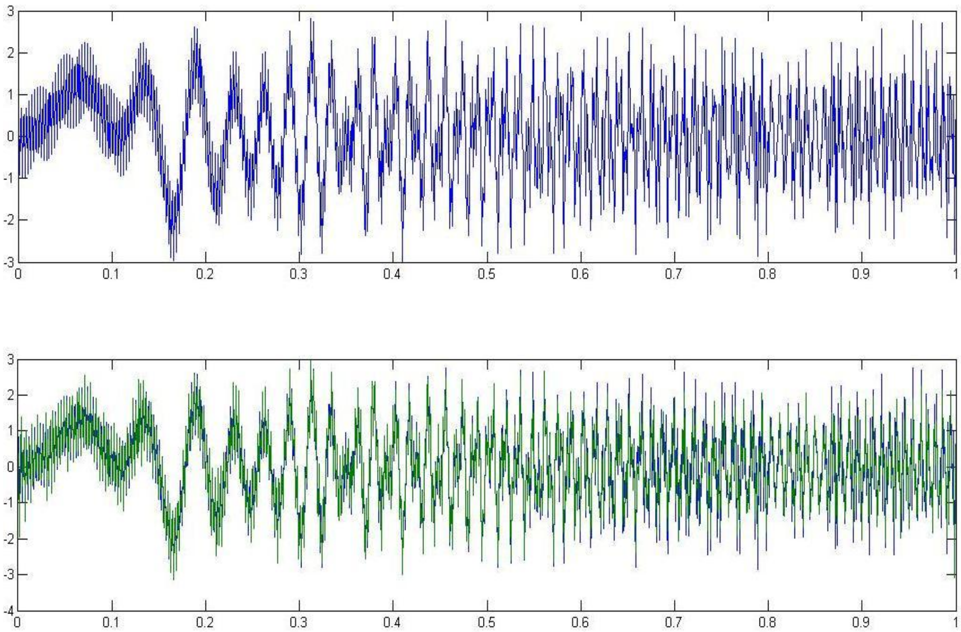


Figure 33: Compression Ratio 0.5 (upper-original and lower-compressed)

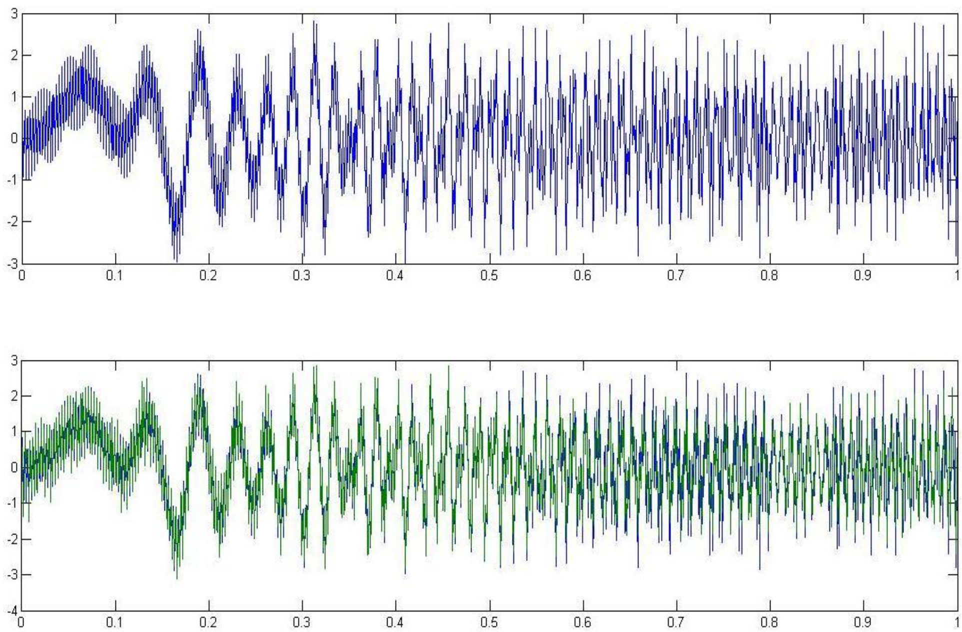


Figure 34: Compression Ratio 0.6 (upper-original and lower-compressed)

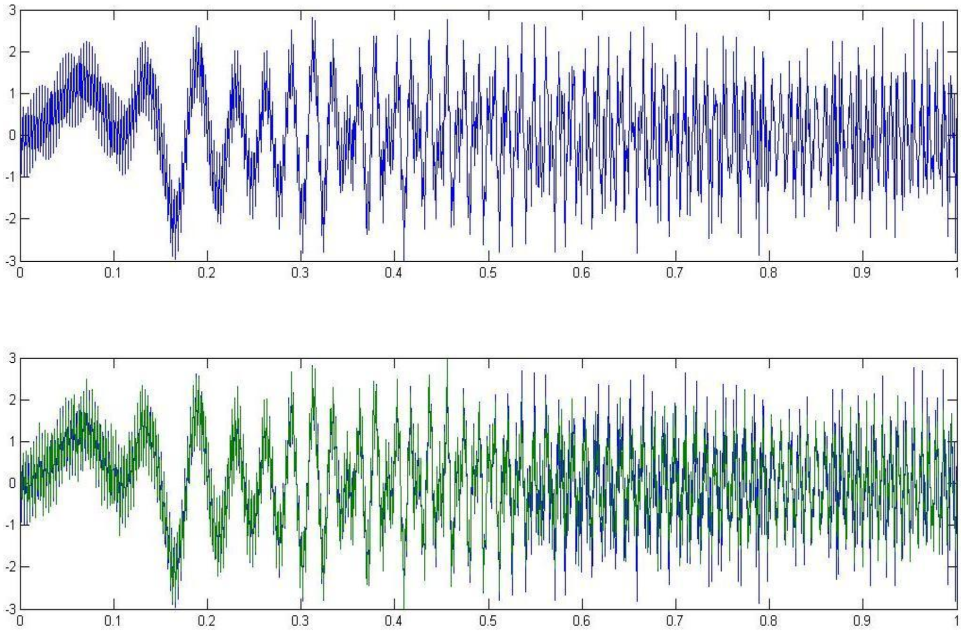


Figure 35: Compression Ratio 0.7 (upper-original and lower-compressed)

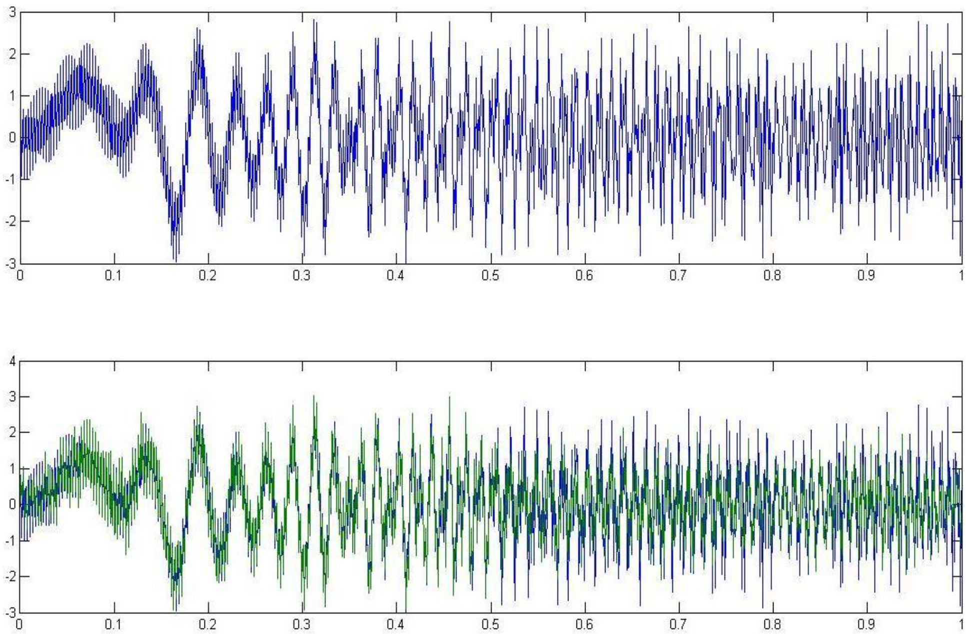


Figure 36: Compression Ratio 0.8 (upper-original and lower-compressed)

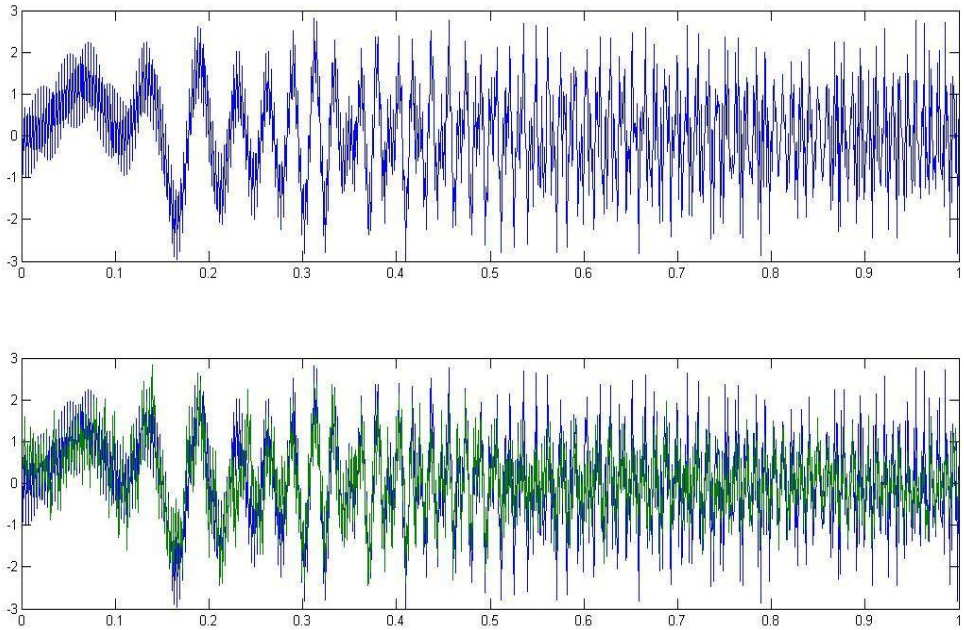


Figure 37: Compression Ratio 0.9 (upper-original and lower-compressed)

From Figure 11 to Figure 37, all compression ratios (0.1 to 0.9) are shown to give clearer observation of the signal changes.

Table 3: Analysis for Signal compression using FFT

CR	SIGNAL 1		SIGNAL 2		SIGNAL 3	
	MSE	RMSE	MSE	RMSE	MSE	RMSE
0.1	4.5956e-4	0.0214	7.9611e-6	0.0028	0.0165	0.1284
0.2	0.0016	0.0406	6.0805e-5	0.0078	0.0421	0.2051
0.3	0.0041	0.0641	1.9570e-4	0.0140	0.0731	0.2704
0.4	0.0080	0.0897	4.3705e-4	0.0209	0.1079	0.3284
0.5	0.0144	0.1200	7.9165e-4	0.0281	0.1461	0.3822
0.6	0.0242	0.1556	0.0012	0.0352	0.1863	0.4317
0.7	0.0384	0.1960	0.0019	0.0431	0.2330	0.4827
0.8	0.0649	0.2547	0.0030	0.0548	0.3170	0.5630
0.9	0.1395	0.3735	0.0062	0.0787	0.5546	0.7447

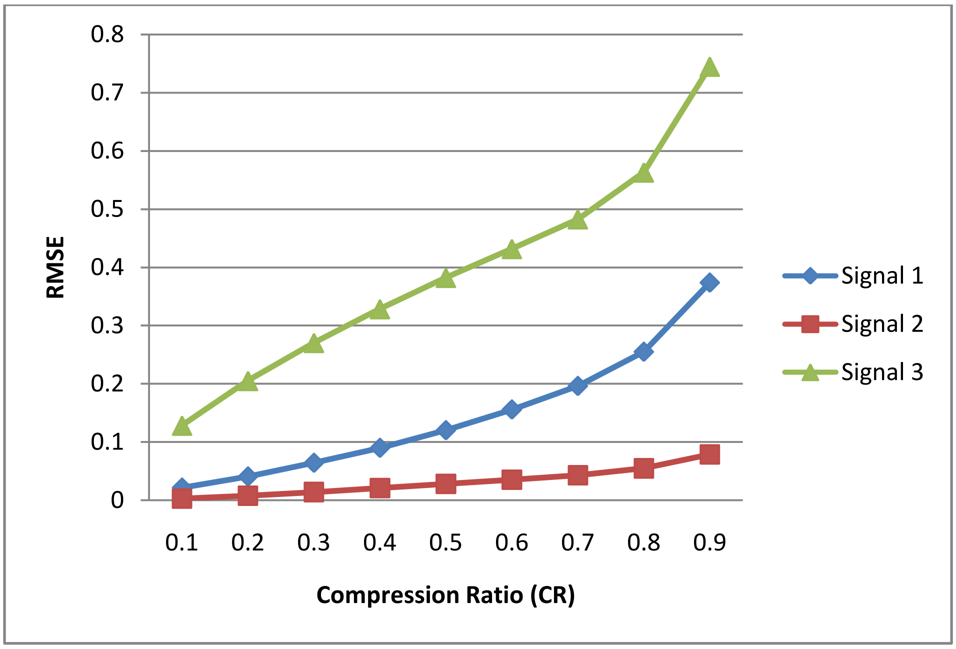


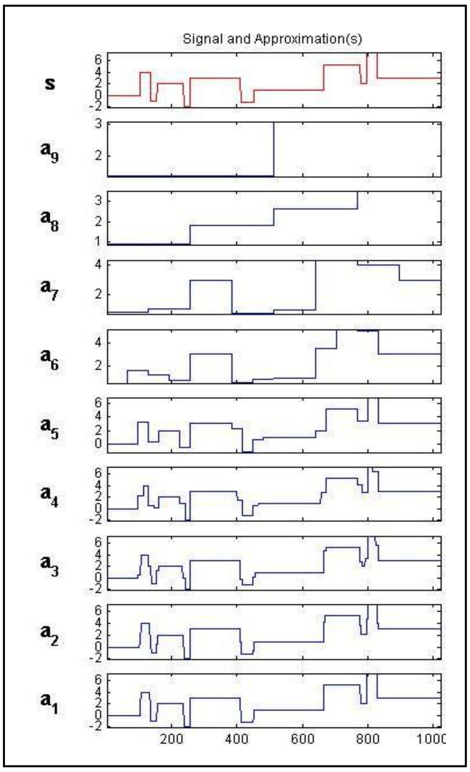
Figure 38: RMSE vs. Compression Ratio

From Table 3, it clearly shows as CR increases, the error also increases. It is true because the higher we remove the number of coefficients, the higher the error will be. For the purpose of comparison with DWT method, only two values of compression ratio are considered (0.8 and 0.9). It is because one of the criteria of a good compression ratio is able to remove huge number of coefficients.

From the above table, it can be seen that among three signals, FFT gives a lowest error for Signal 2 (“Heavy Sine”) and the highest error for Signal 3 (“Mishmash”) but it is not a solid reason to declare FFT is the best for Signal 2 till comparison is made with DWT method.

4.2 Signal Compression Using Discrete Wavelet Transform (DWT)

4.2.1 Signal 1("Block")



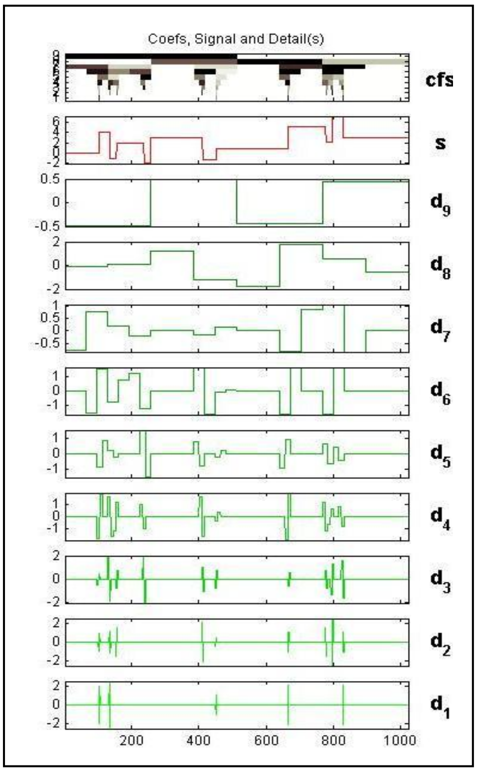


Figure 39a: Signal and Approximations by using Haar (D2)

Figure 39b: Signal and Details by using Haar (D2)

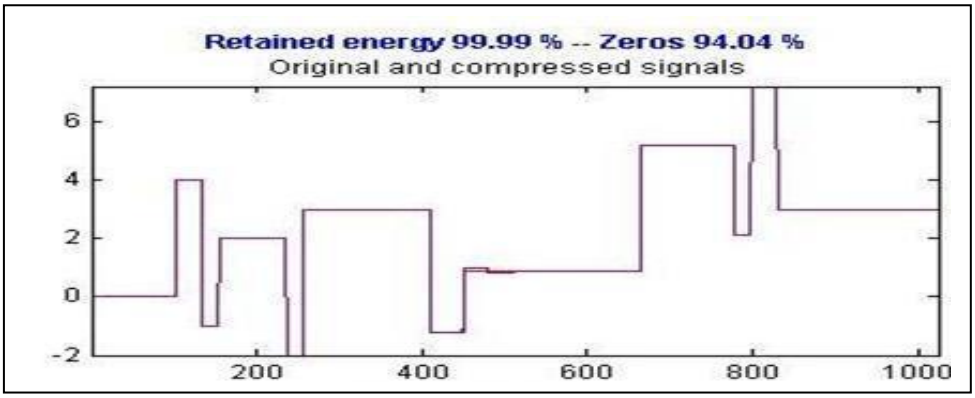


Figure 39c: Original and Compressed Signal

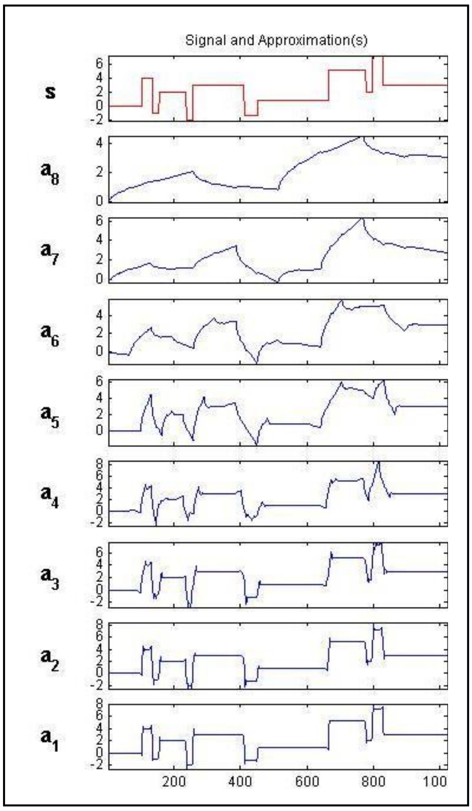


Figure 40a: Signal and Approximations by using D4

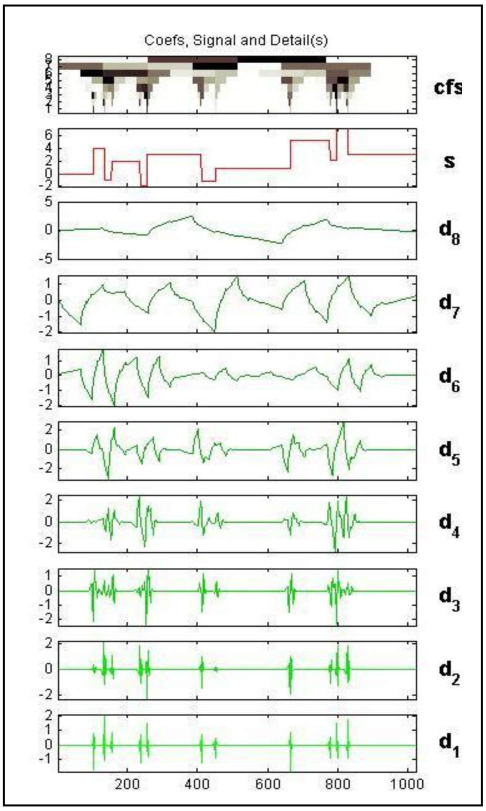


Figure 40b: Signal and Details by using D4

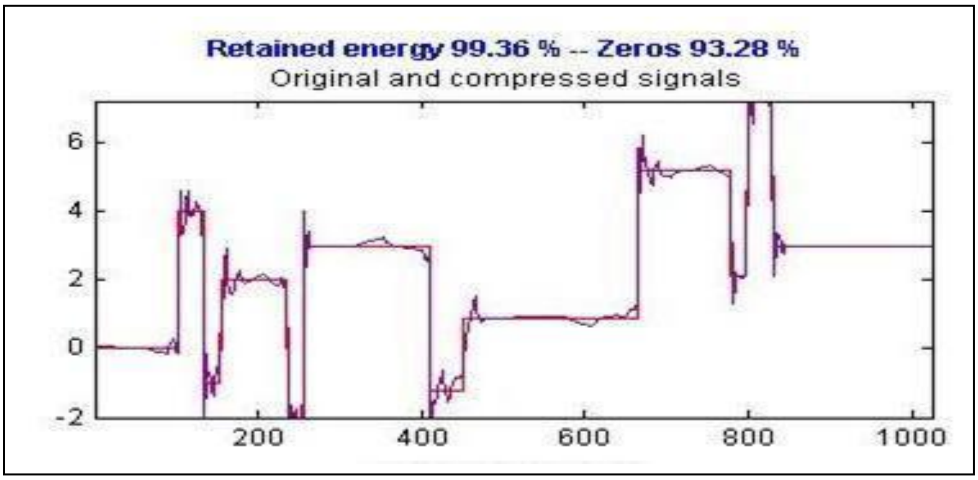


Figure 40c: Original and Compressed Signal

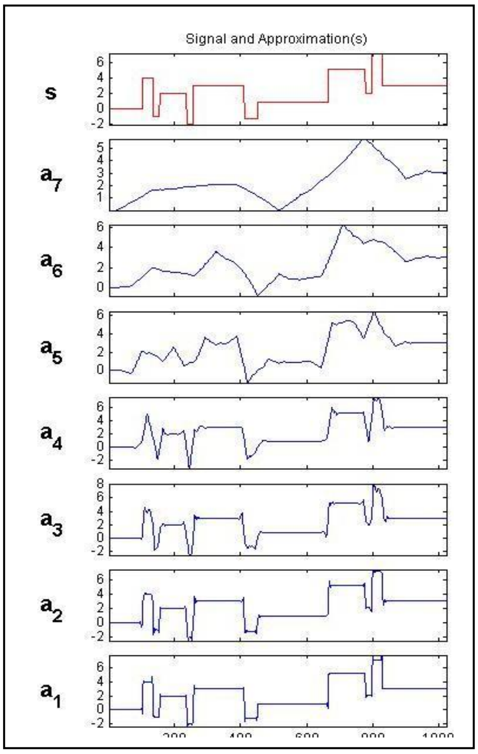


Figure 41a: Signal and Approximations by using D6

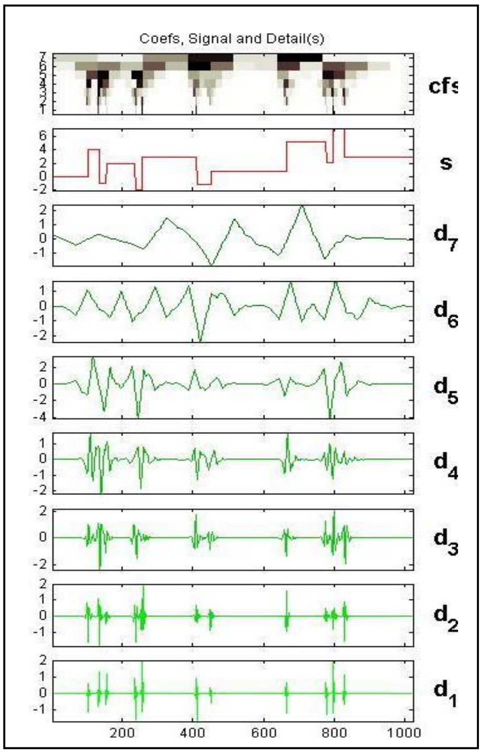


Figure 41b: Signal and Details by using D6

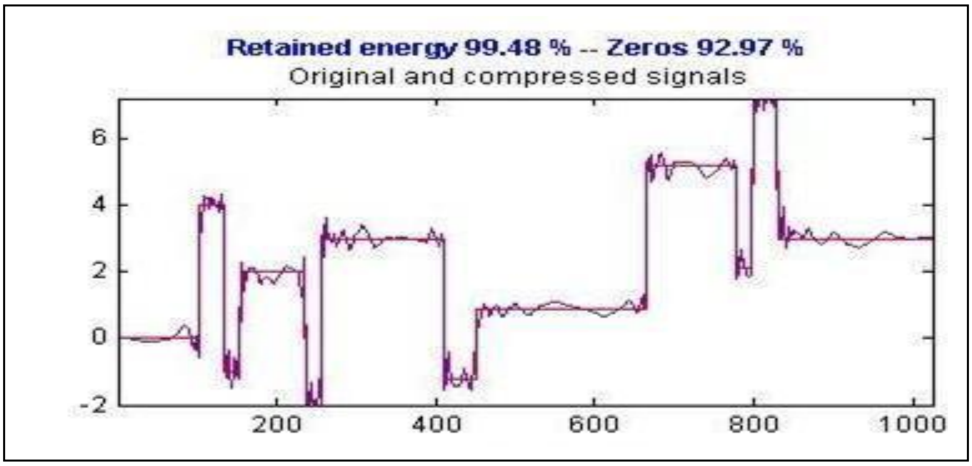


Figure 41c: Original and Compressed Signal

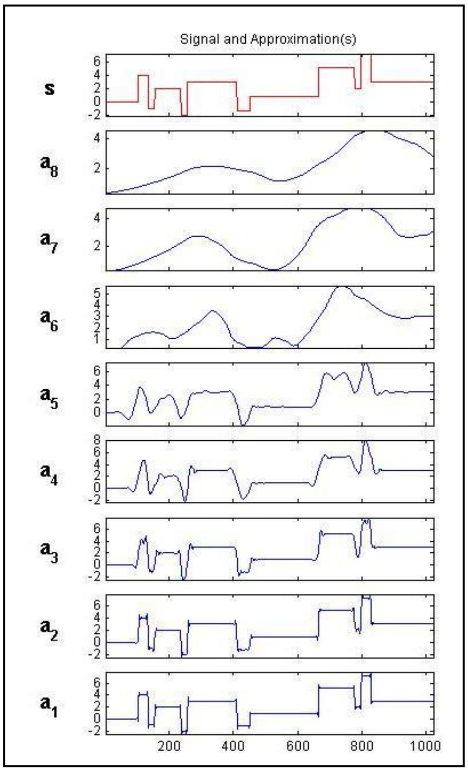


Figure 42a: Signal and Approximations by using D8

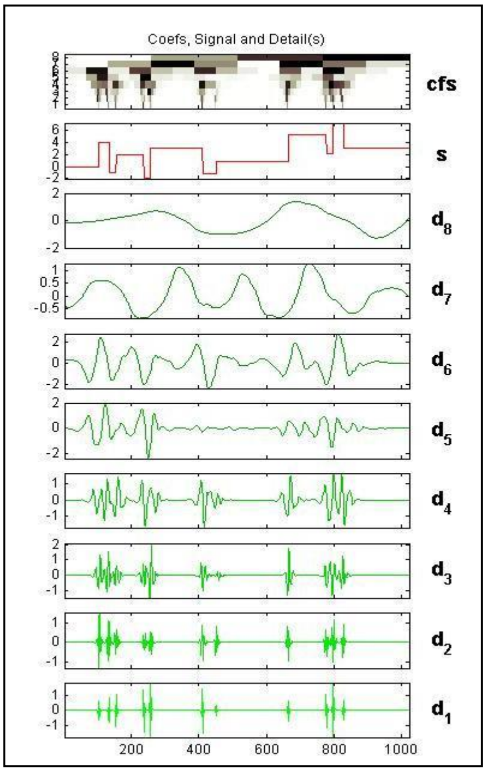


Figure 42b: Signal and Details by using D8

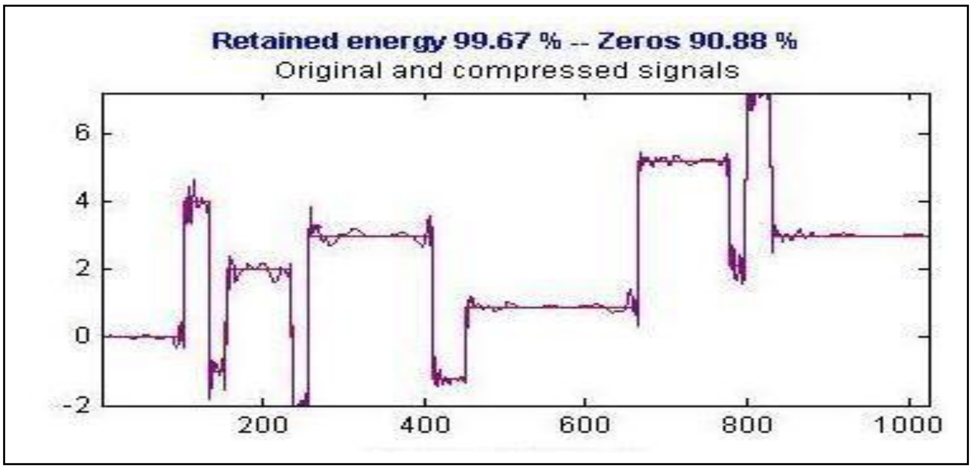
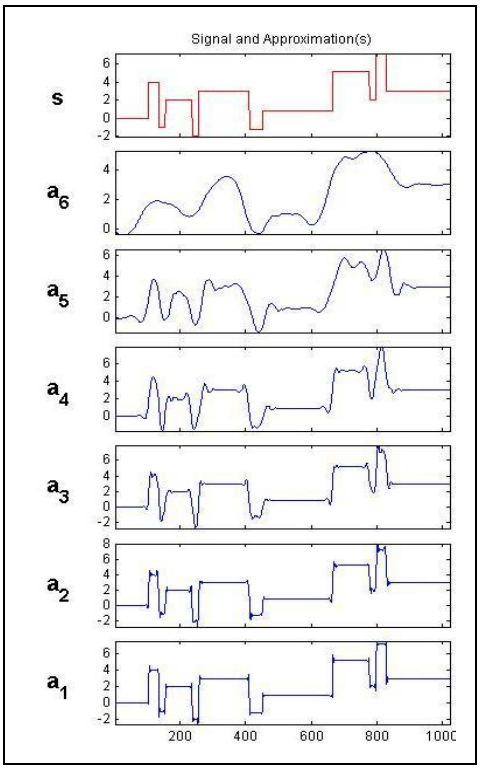


Figure 42c: Original and Compressed Signal



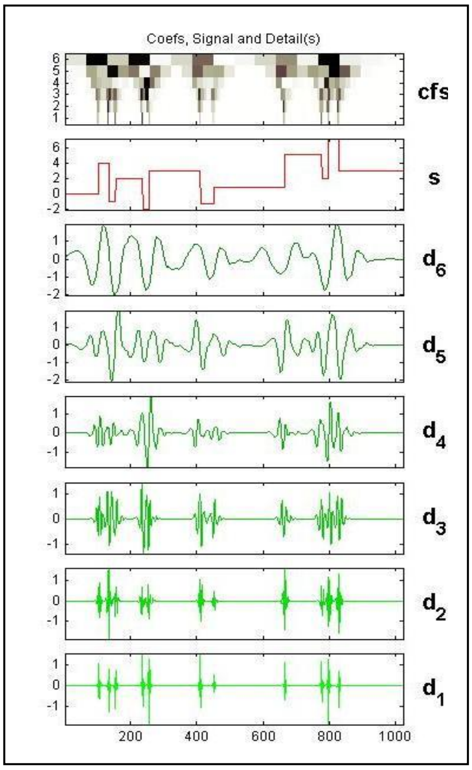


Figure 43a: Signal and Approximations by using D10

Figure 43b: Signal and Details by using D10

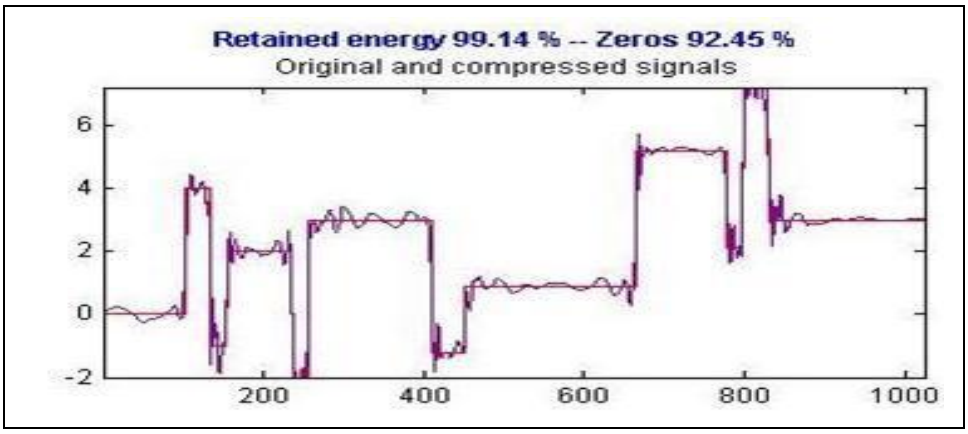


Figure 43c: Original and Compressed Signal

Table 4: Results Summary of Signal 1 ("Block")

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	9	2:1	0	0
D4	8	4:1	0.0563	0.2373
D6	7	8:1	0.0457	0.2137
D8	8	4:1	0.0326	0.1805
D10	6	16:1	0.0794	0.2817

From Table 4, it is very obvious that Haar is the best filter for Signal 1 ("Block"). The MSE and RMSE are both zero. For other filters, the measured errors are quite large.

Table 5: Comparison of Daubechies Family for Same Maximum Level

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	9	2:1	0	0
D4	9	2:1	0.0942	0.3070
D6	9	2:1	0.0559	0.2364
D8	9	2:1	0.0762	0.2760
D10	9	2:1	0.1563	0.3954

For the same level of decomposition, Haar still the best compared to other filters. So, considering wavelet only, Haar is incomparable to other Daubechies filters for the same level with MSE and RMSE both are almost zero.

Table 6: Overall Comparison for Signal 1 (“Block”)

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	9	2:1	0	0
D4	8	4:1	0.0563	0.2373
D6	7	8:1	0.0457	0.2137
D8	8	4:1	0.0326	0.1805
D10	6	16:1	0.0794	0.2817
FFT1	-	5:1	0.0649	0.2547
FFT2	-	10:1	0.1395	0.3735

From Table 6, based on the error values (MSE and RMSE), Haar gives the lowest possible value. In term of compression ratio, D6, D10 and FFT2 are among the best with high compression ratio but considering the error as well, D10 is the best among three of them. For overall, Haar is the best filter for Signal 1 (“Block”) because the main parameter that we consider first is error.

4.2.2 Signal 2 (“Heavy Sine”)

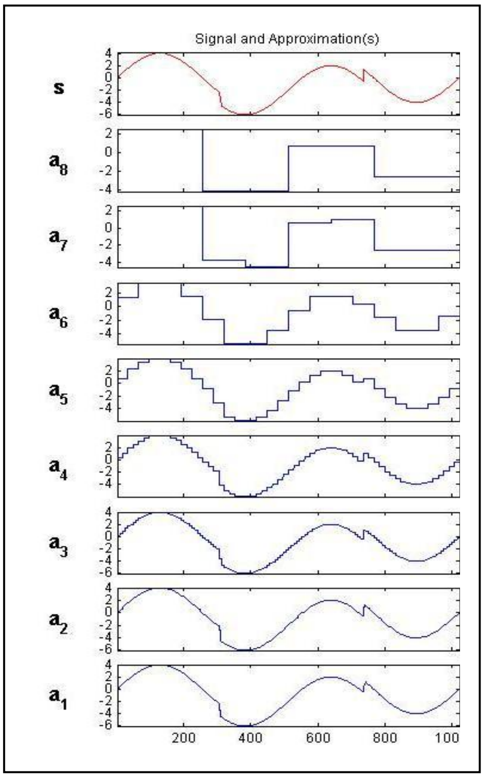


Figure 44a: Signal and Approximations by using Haar (D2)

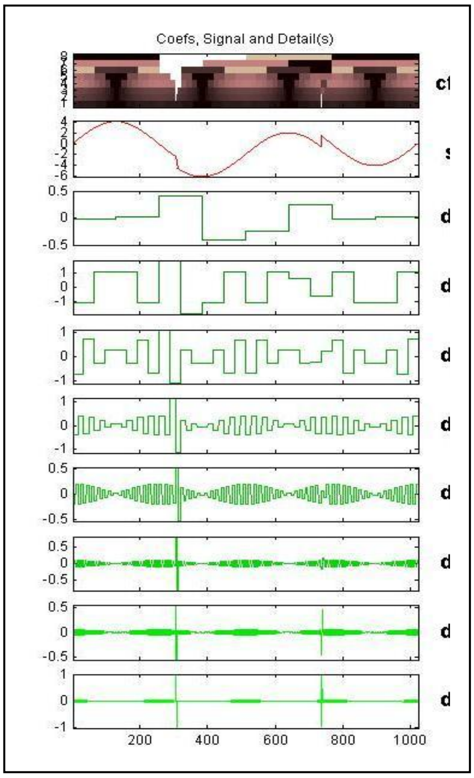


Figure 44b: Signal and Details by using Haar (D2)

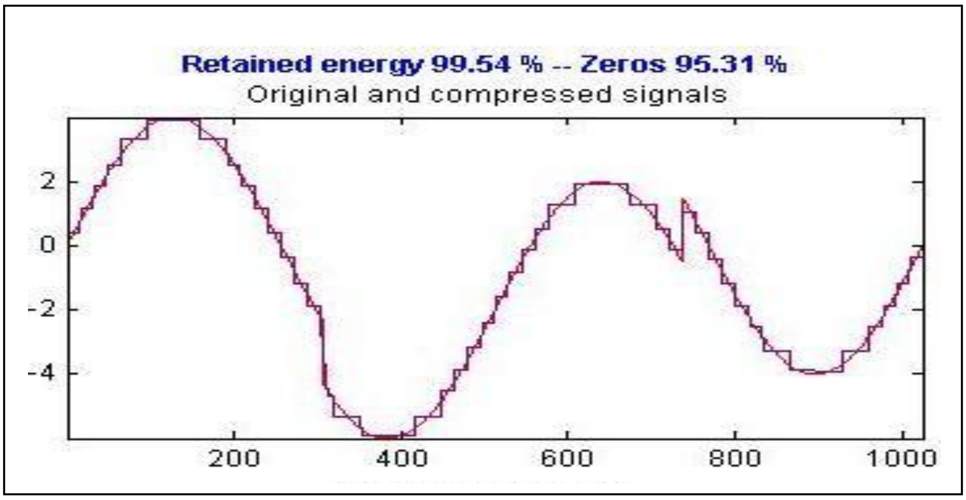
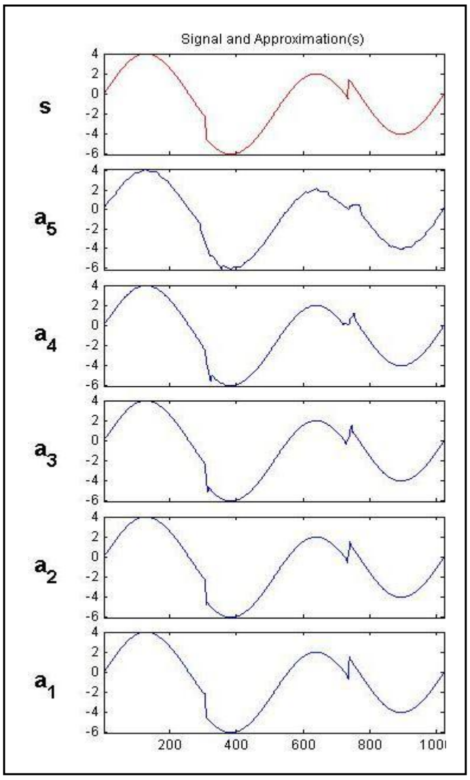


Figure 44c: Original and Compressed Signal



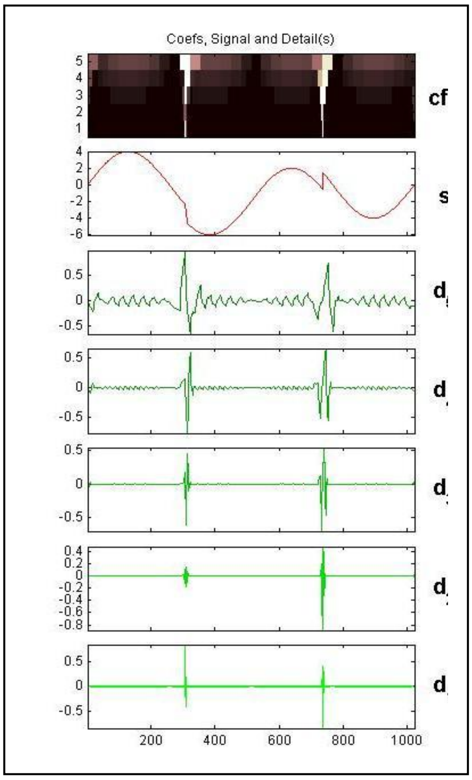


Figure 45a: Signal and Approximations by D4

Figure 45b: Signal and Details by using D4

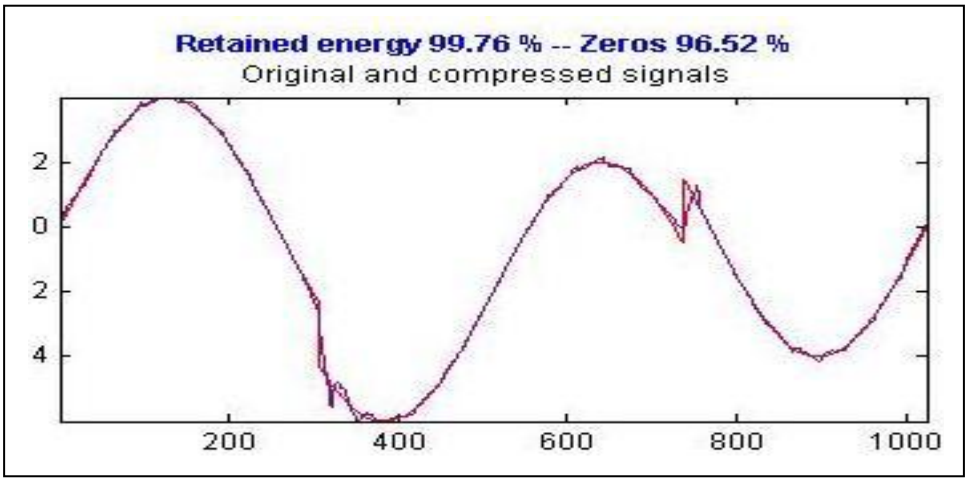


Figure 45c: Original and Compressed Signal

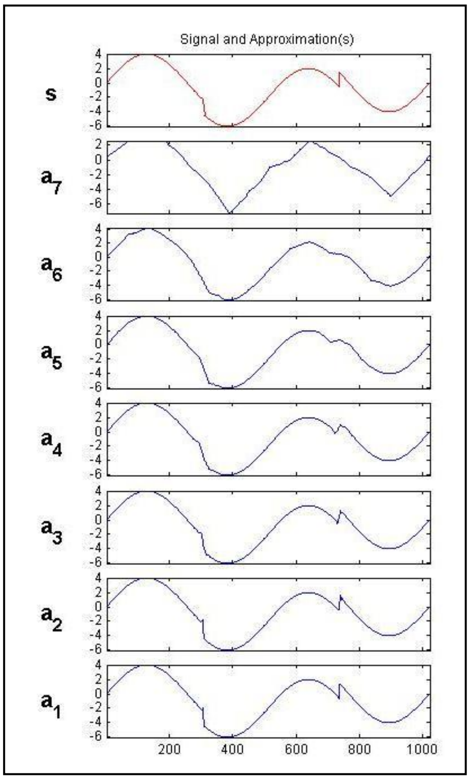


Figure 46a: Signal and Approximations by using D6

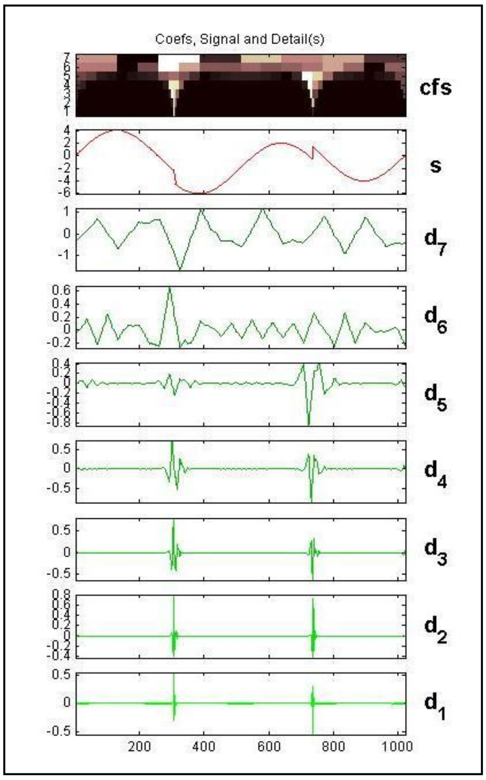


Figure 46b: Signal and Details by using D6

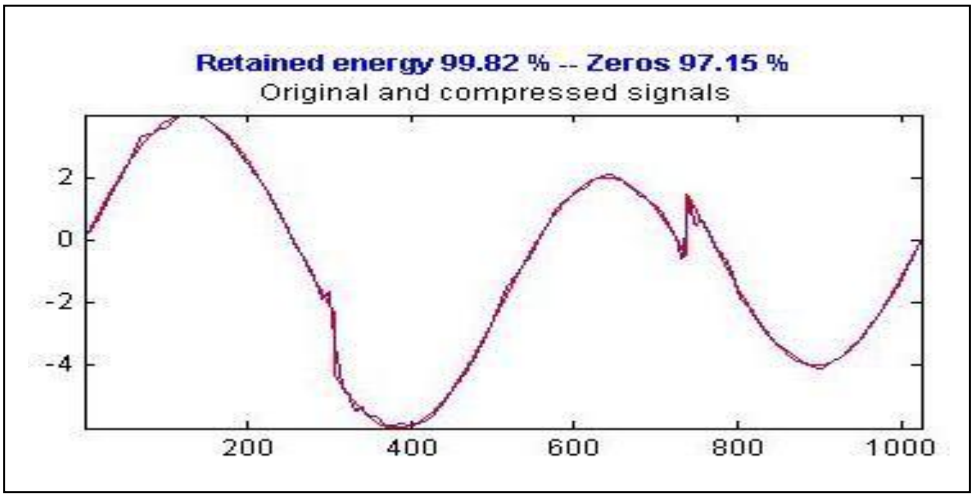


Figure 46c: Original and Compressed Signal

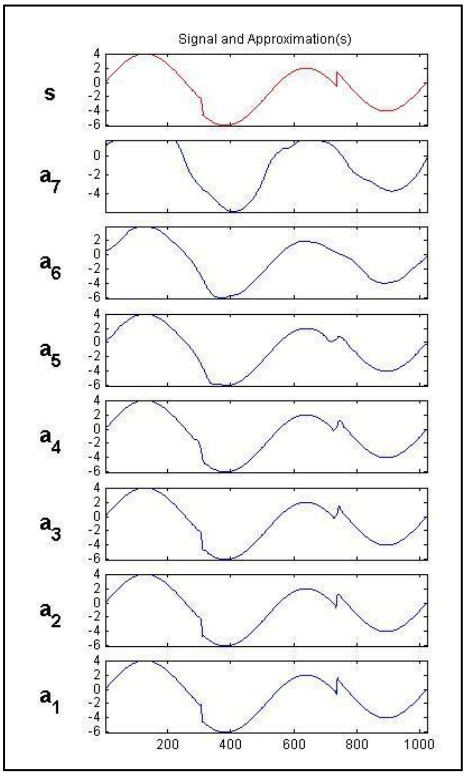


Figure 47a: Signal and Approximations by using D8

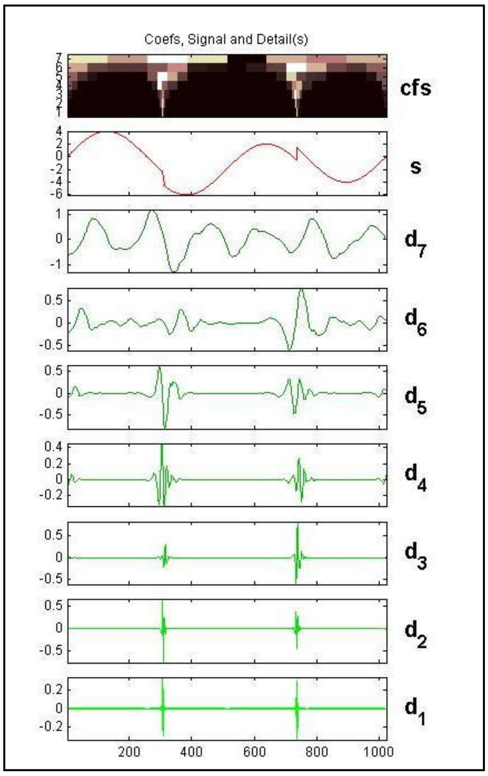


Figure 47b: Signal and Details by using D8

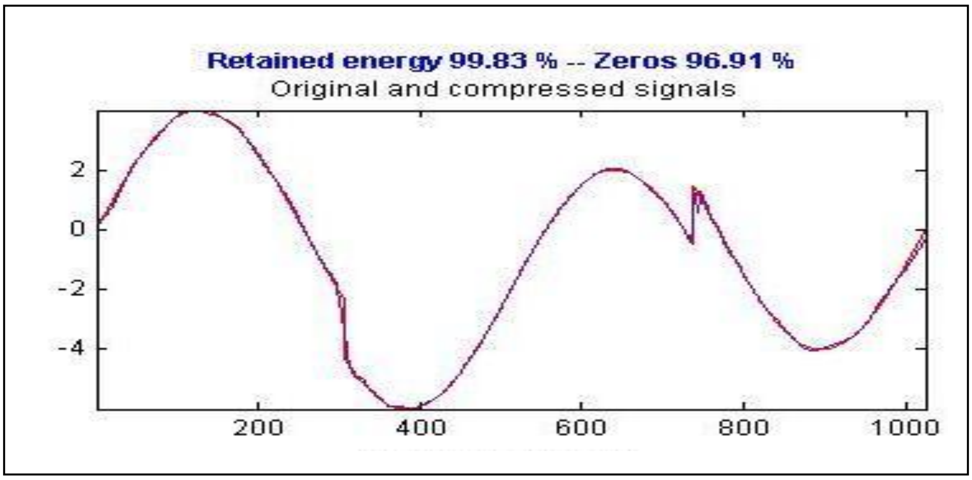
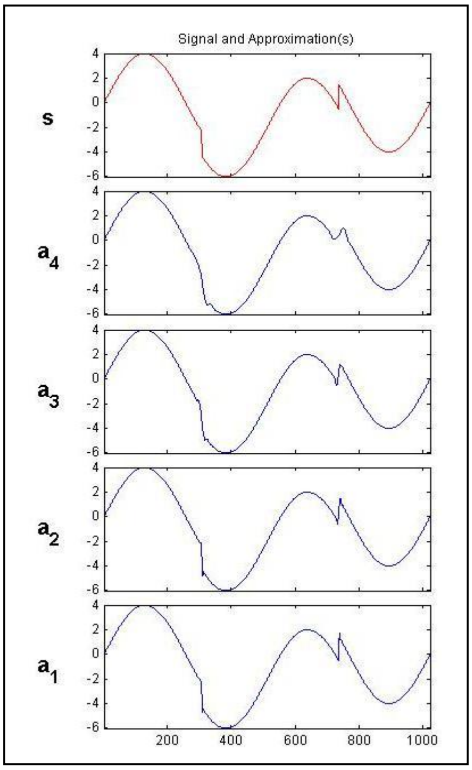


Figure 47c: Original and Compressed Signal



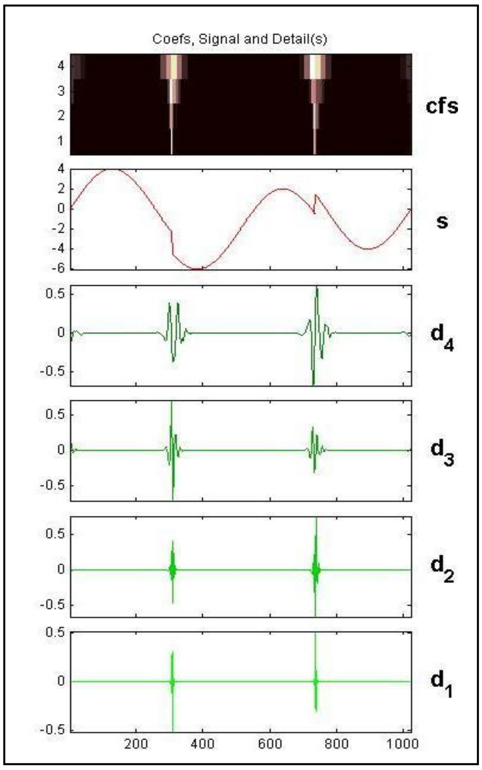


Figure 48a: Signal and Approximations by using D10

Figure 48b: Signal and Details by using D10

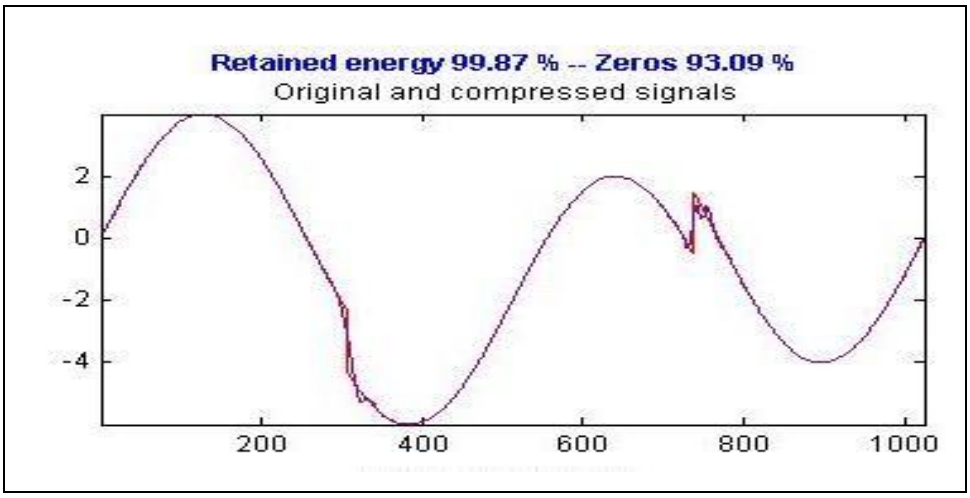


Figure 48c: Original and Compressed Signal

Table 7: Result Summary for Signal 2("Heavy Sine")

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	8	4:1	0.0421	0.2051
D4	5	32:1	0.0189	0.1376
D6	7	8:1	0.0147	0.1211
D8	7	8:1	0.0107	0.1036
D10	4	64:1	0.0098	0.0989

From Table 7, all Daubechies filters give almost the same error but not for Haar (D2). Then, by looking at second parameter which is compression ratio, D10 give the highest possible ratio which is 64:1. So, for DWT on Signal 2, D10 give the best result.

Table 8: Comparison of Daubechies Family for Same Maximum Level

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	7	8:1	0.0473	0.2174
D4	7	8:1	0.0228	0.1511
D6	7	8:1	0.0147	0.1211
D8	7	8:1	0.0107	0.1036
D10	7	8:1	0.0177	0.1332

For the same level of decomposition, D8 is a bit better compared to D10 in term of errors (MSE and RMSE). Overall, the performance of Daubechies filters is very good for this type of signal except for Haar (D2).

Table 9: Overall Comparison for Signal 2 ("Heavy Sine")

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	8	4:1	0.0421	0.2051
D4	5	32:1	0.0189	0.1376
D6	7	8:1	0.0147	0.1211
D8	7	8:1	0.0107	0.1036
D10	4	64:1	0.0098	0.0989
FFT1	-	5:1	0.0030	0.0548
FFT2	-	10:1	0.0062	0.0787

From Table 9, after comparing DWT and FFT together, the results agree that FFT method is outperformed the ability of DWT for this type of signal. The calculated MSE and RMSE show for FFT1 and FFT2, both give the lowest possible value. So, for Signal 2 ("Heavy Sine"), FFT performed better than DWT.

4.2.3 Signal 3 (“Mishmash”)

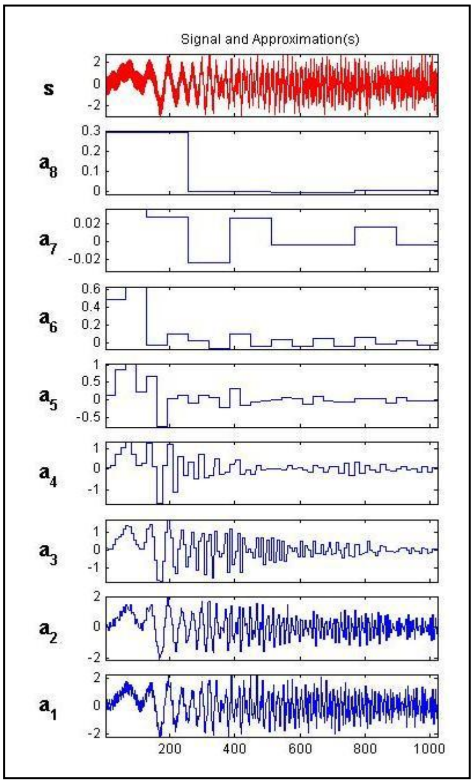


Figure 49a: Signal and Approximations by using Haar (D2)

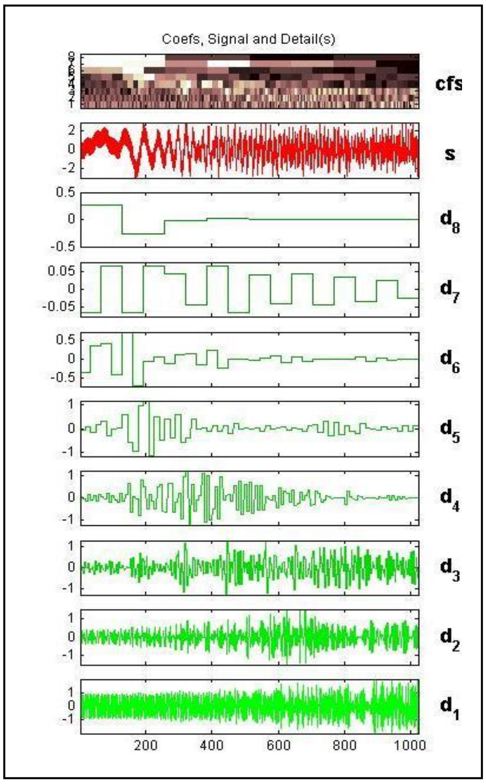


Figure 49b: Signal and Details by using Haar (D2)

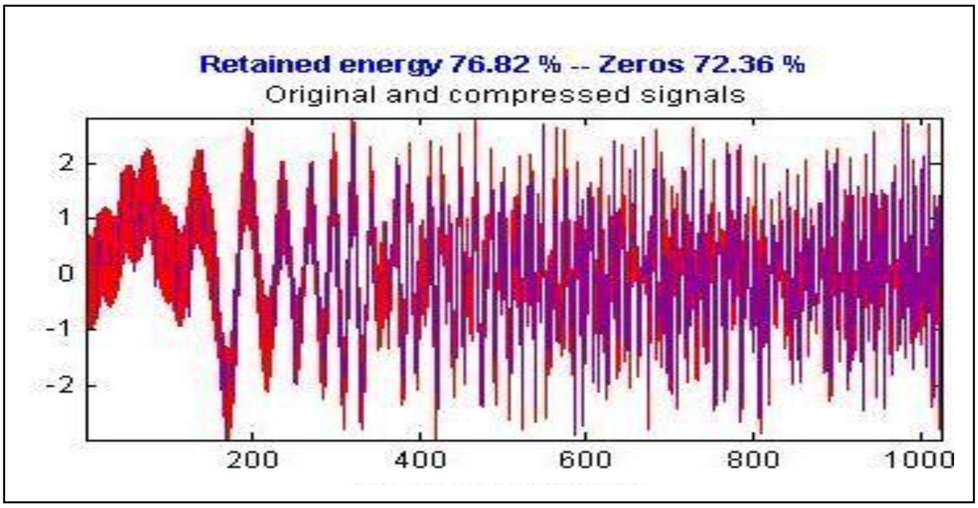


Figure 49c: Original and Compressed Signal

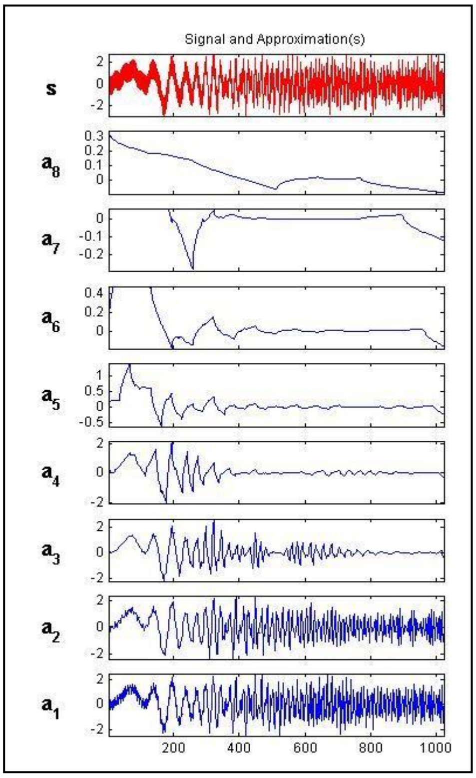


Figure 50a: Signal and Approximations by using D4

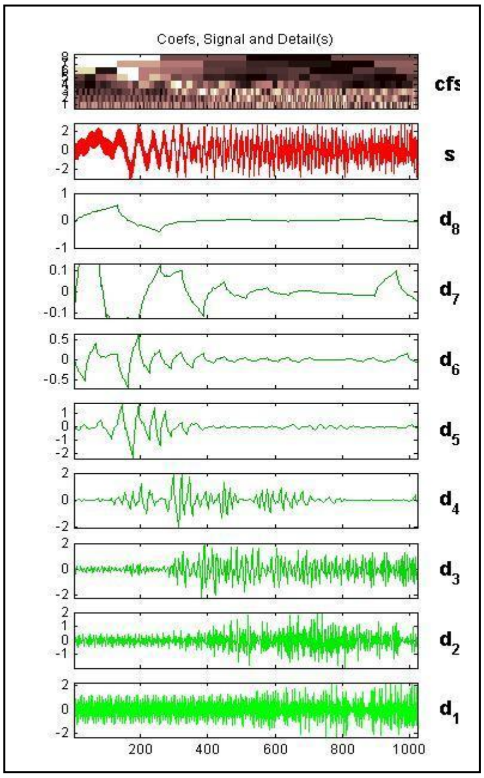


Figure 50b: Signal and Details by using D4

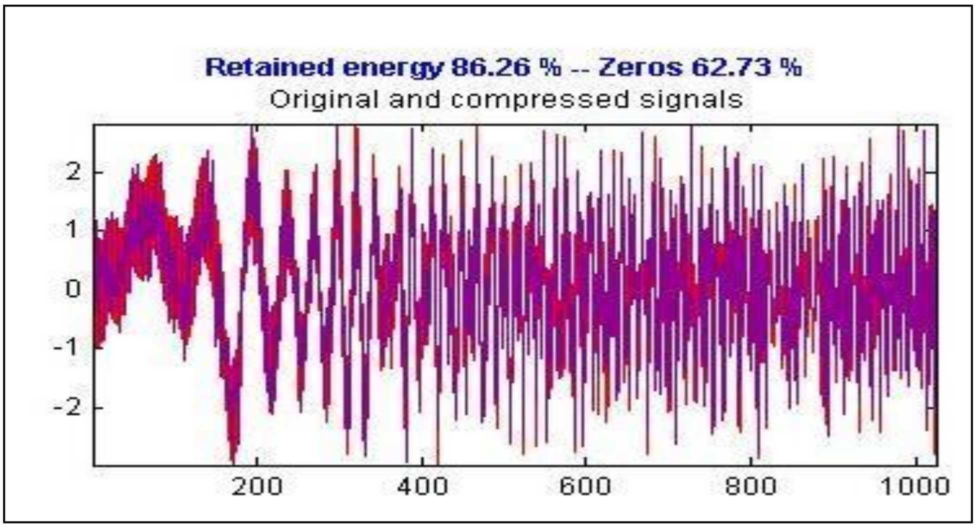


Figure 50c: Original and Compressed Signal

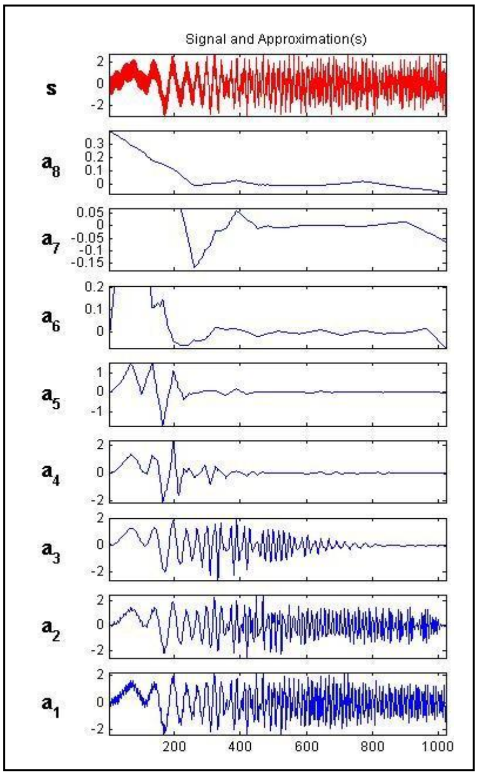


Figure 51a: Signal and Approximations by using D6

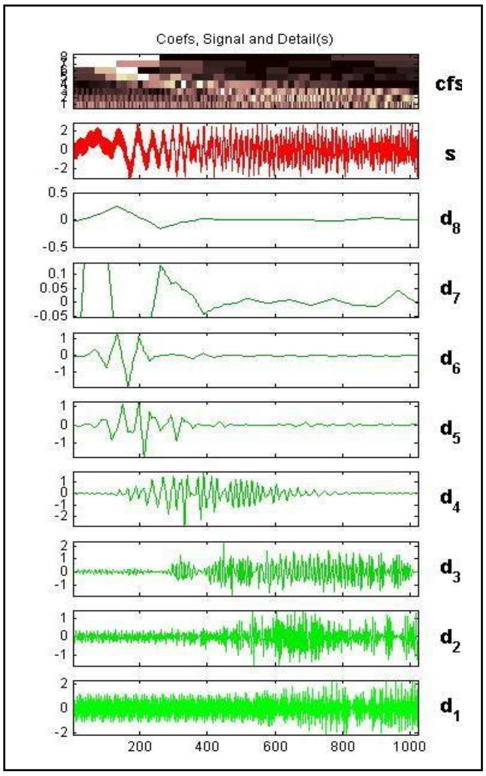


Figure 51b: Signal and Details by using D6

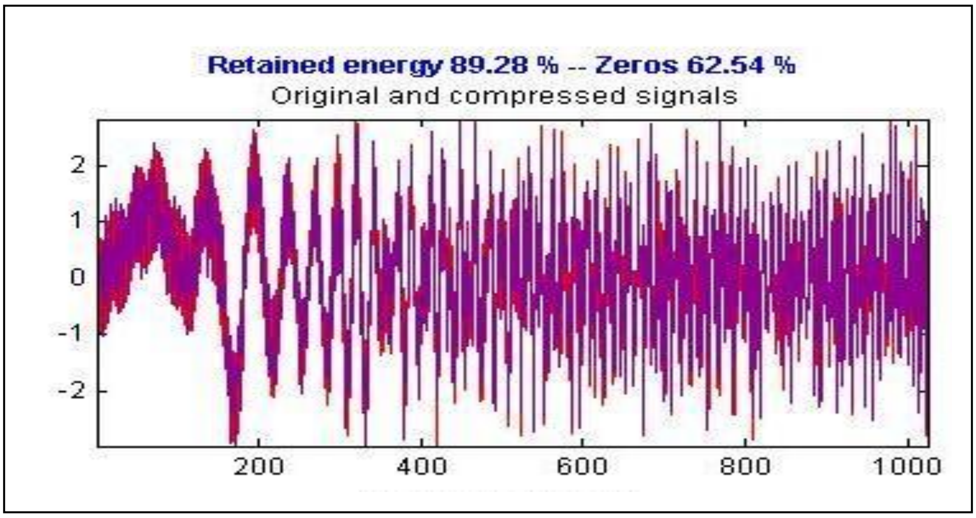


Figure 51c: Original and Compressed Signal

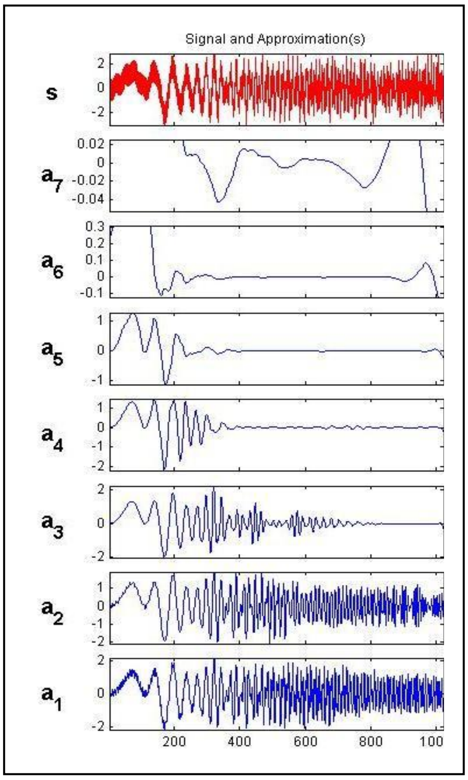


Figure 52a: Signal and Approximations by using D8

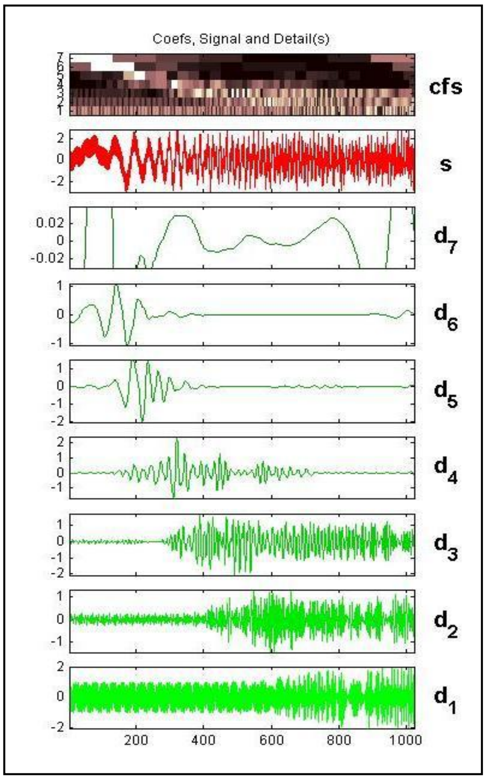


Figure 52b: Signal and Details by using D8

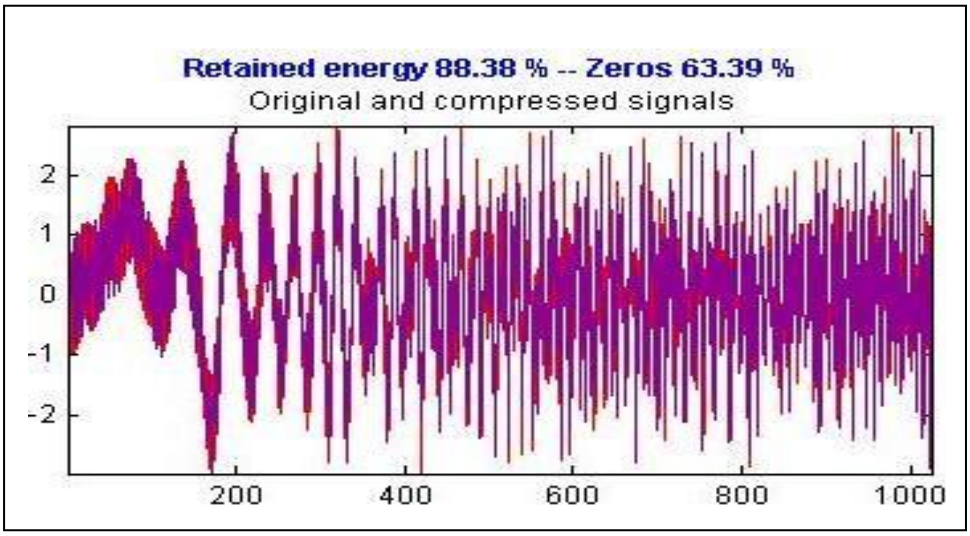
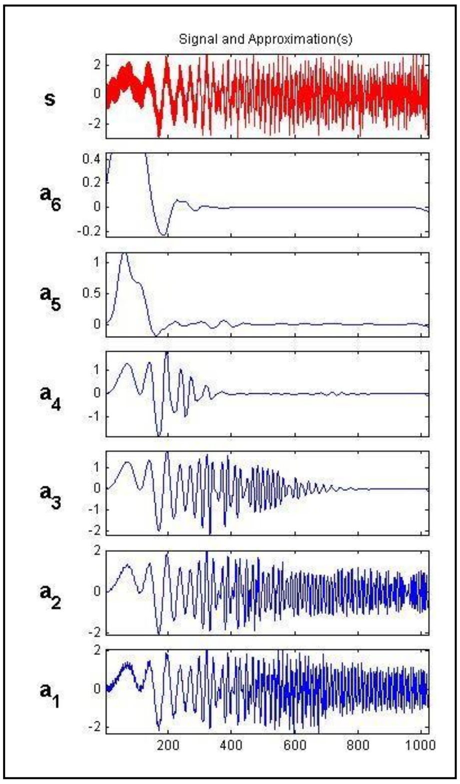


Figure 52c: Original and Compressed Signal



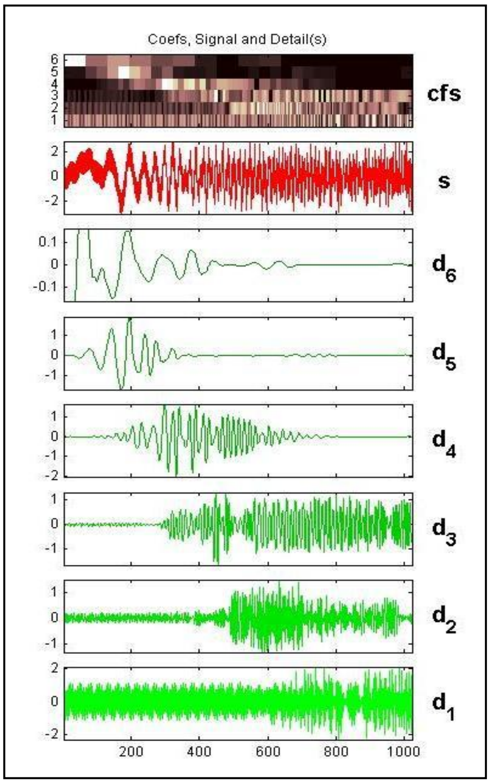


Figure 53a: Signal and Approximations by using D10

Figure 53b: Signal and Details by using D10

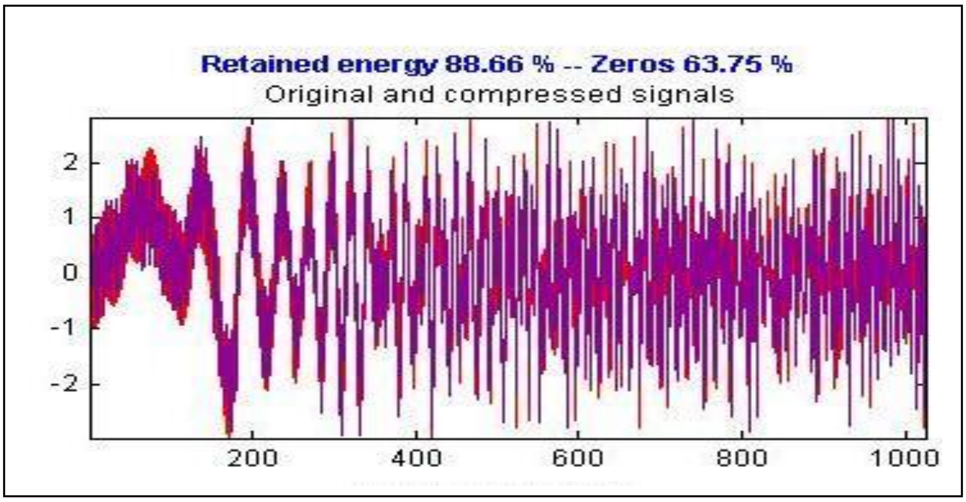


Figure 53c: Original and Compressed Signal

Table 10: Result Summary for Signal 32("Mishmash")

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	8	4:1	0.3369	0.5804
D4	8	4:1	0.1990	0.4461
D6	8	4:1	0.1627	0.4034
D8	7	8:1	0.1698	0.4121
D10	6	16:1	0.1642	0.4052

From Table 10, there are 3 filters which give lowest error which are D6, D8 and D10. By looking at second parameter (compression ratio), D10 is the best among them because it give among the lowest error but with the highest compression ratio.

Table 11: Comparison of Daubechies Family for Same Maximum Level

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	7	8:1	0.3367	0.5803
D4	7	8:1	0.4181	0.6466
D6	7	8:1	0.1626	0.4032
D8	7	8:1	0.1698	0.4121
D10	7	8:1	0.1645	0.4055

For the same optimum level, D6 is slightly better than D10 even though the error for them is almost the same including the D8. Overall, D6, D8 and D10 are among the filters that are good for Signal 3.

Table 12: Overall Comparison between FFT and DWT

	Optimum Level	Compression Ratio	MSE	RMSE
Haar	8	4:1	0.3369	0.5804
D4	8	4:1	0.1990	0.4461
D6	8	4:1	0.1627	0.4034
D8	7	8:1	0.1698	0.4121
D10	6	16:1	0.1642	0.4052
FFT1	-	5:1	0.3170	0.5630
FFT2	-	10:1	0.5546	0.7447

For overall comparison, the performance of Daubechies filters (except Haar) is really good compared to FFT. From the calculated errors, D6, D8 and D10 give the lowest possible error which is less than 0.2 for MSE and less than 0.5 for RMSE. Individually, D10 is the best due to high compression ratio. Overall, for this type of signal, Daubechies is really strong compared to FFT.

4.3 Image Compression Using Discrete Wavelet Transform

For this part, there are 4 images (see Figure 54) were used where they are the universal or common images used by researchers. For each image, three fixed compression ratio were chosen which are 5:1, 10:1 and 20:1 so at the end the comparison will be made based on RMSE and Peak to Signal Noise Ratio (PSNR) for the same compression ratio. In addition, the filters used for this part are the same as in signal compression (Haar, D2, D4, D6, D8 and D10) and wavelet decomposition is fixed to level 3 only because this level is already optimum for image compression. The main focus of this part is to observe the effect of the changes of threshold values for each level of decomposition. For a fair comparison, all images are 256x256 matrix and grayscale.

Figure 54a: Image “Lena”

Figure 54b: Image “Peppers”

Figure 54c: Image “House”

Figure 54d: Image “Cameraman”

4.3.1 *Image 1 (“Lena”)*

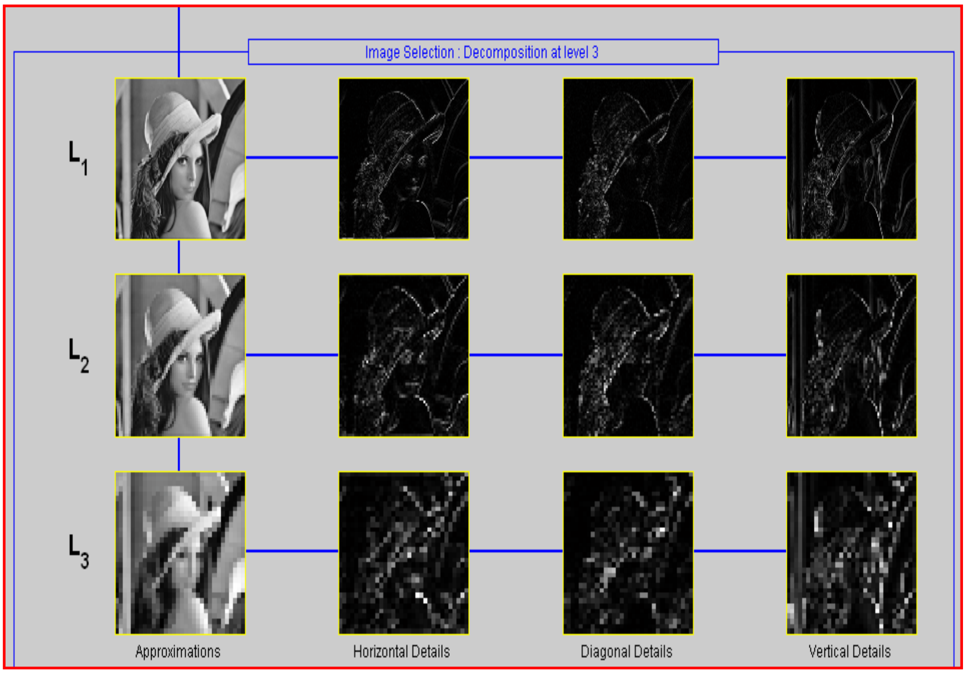


Figure 55a: Image 1 decomposition at level 3 using Haar

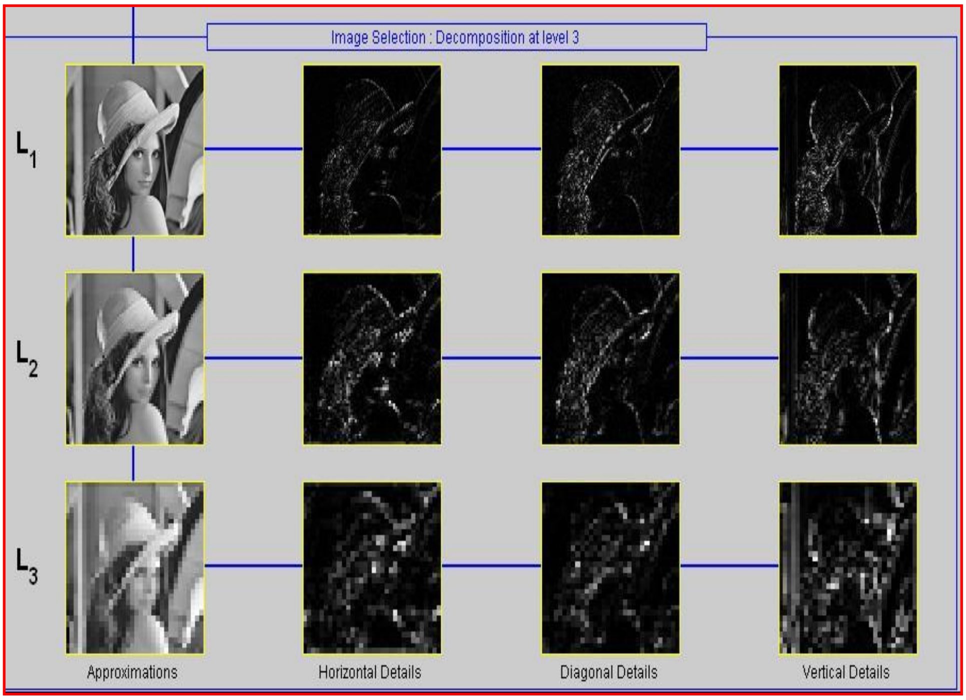


Figure 55b: Image 1 decomposition at level 3 using D4

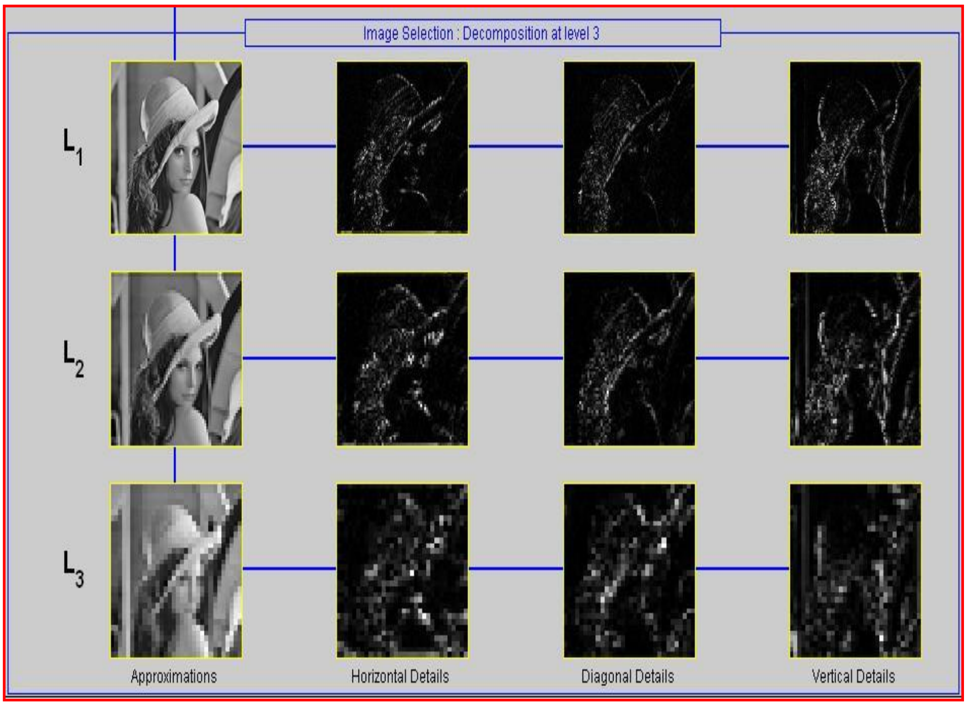


Figure 55c: Image 1 decomposition at level 3 using D6

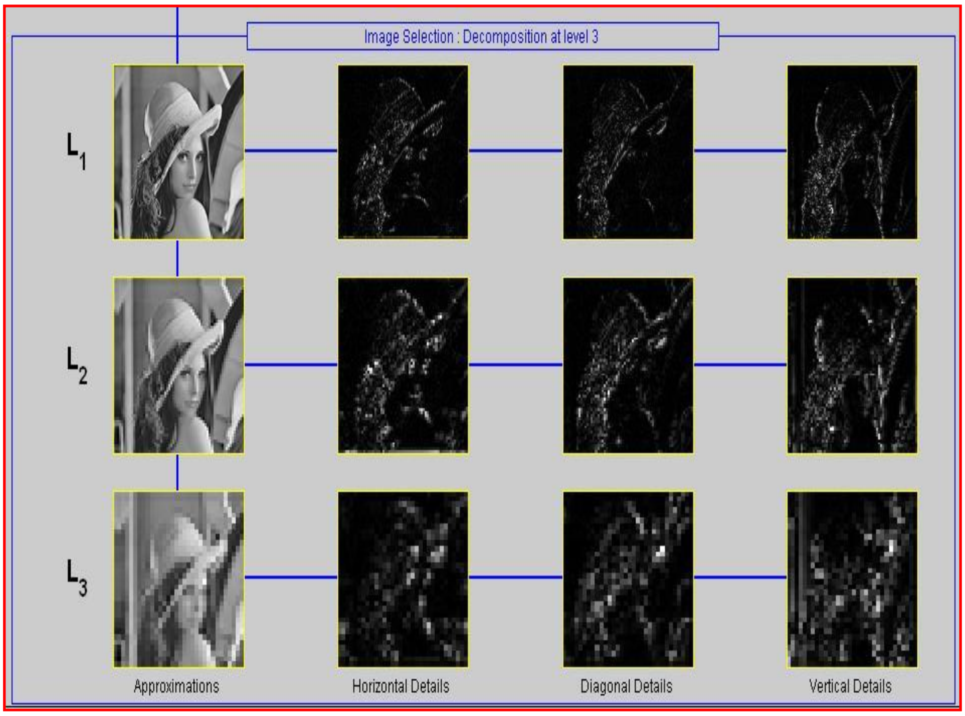


Figure 55d: Image 1 decomposition at level 3 using D8

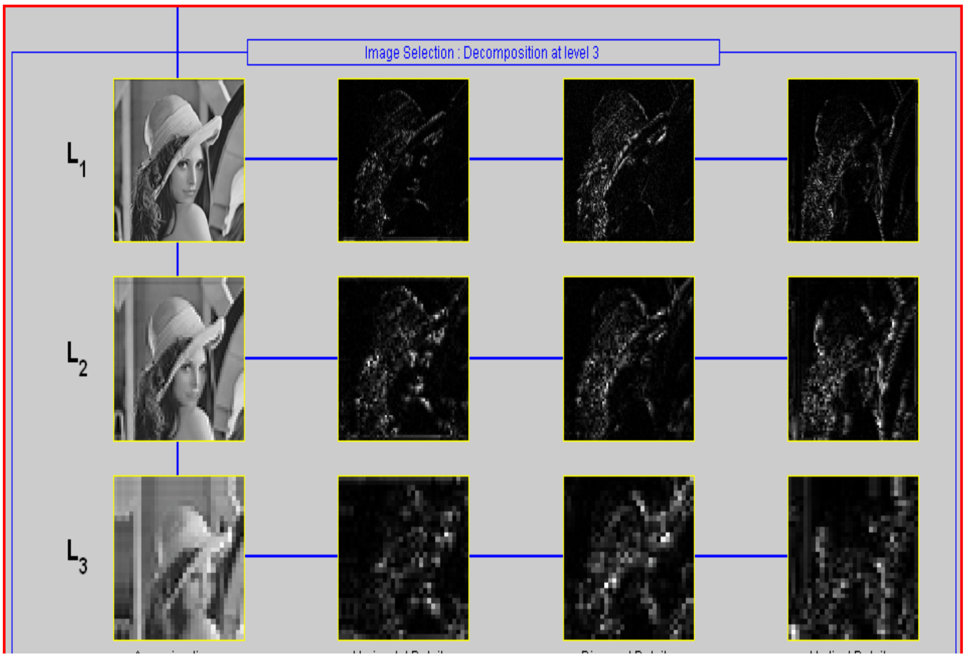


Figure 55e: Image 1 decomposition at level 3 using D10

Figure 56a: Image 1 compression using Haar

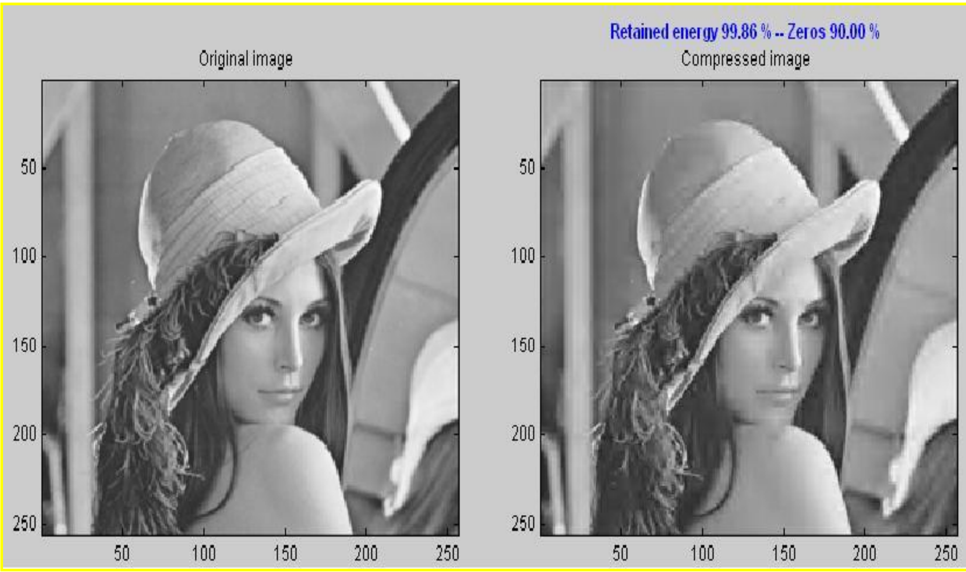


Figure 56b: Image 1 compression using D4

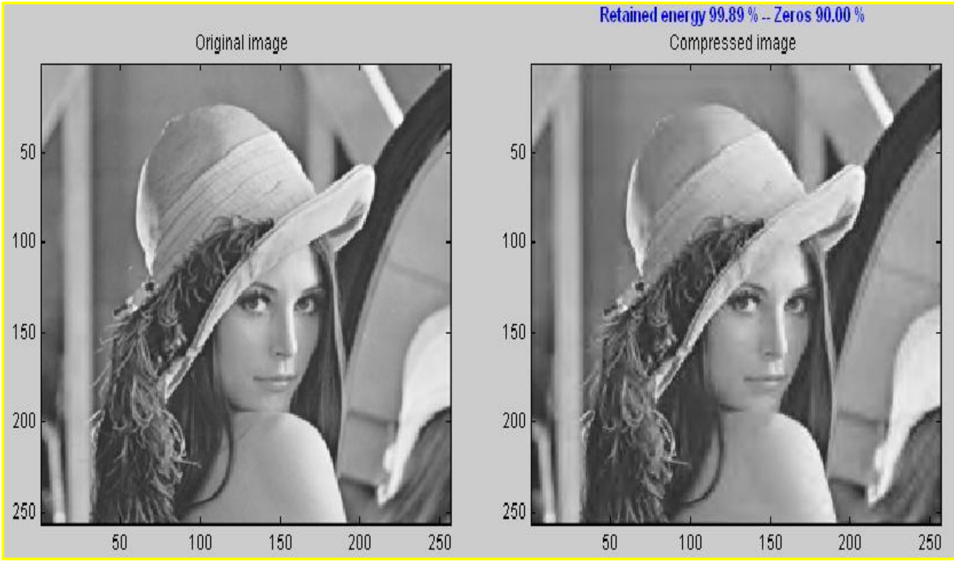


Figure 56c: Image 1 compression using D6

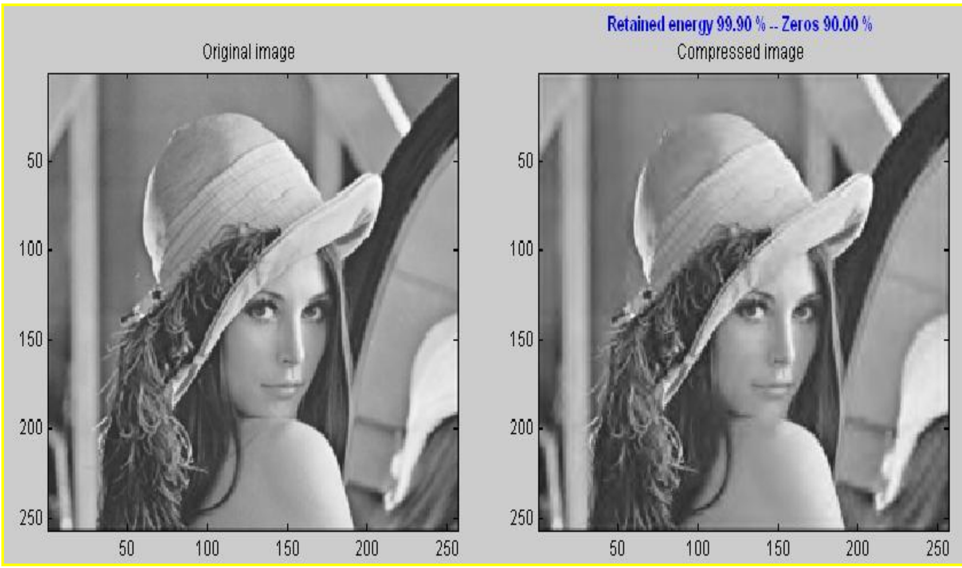


Figure 56d: Image 1 compression using D8

Figure 56e: Image 1 compression using D10

Figure 55 shows the wavelet decomposition at level three using 5 different filters. Figure 56a shows the original and compressed images using Haar D4, D6, D8 and D10 respectively. The purpose of showing these figures is to give a clear observation that Haar is different with other Daubechies filters although Haar is one of the Daubechies type. For wavelet decomposition, from an original image, wavelet decomposes it into 3 parts (horizontal, vertical and diagonal). Level 3 decomposition means wavelet decomposes an image three times.

Table 13: Analysis at CR=5:1, using D8 at level 3

RE(%)	99.85	99.88	99.85	99.86	99.84	99.85	99.85
NOZ(%)	80	80	80	80	80	80	80
Threshold Horizontal (H)	4.145 10.02 43.04	4.145 10.02 43.04	4.145 10.02 43.04	5 12 45	3.5 8.8 41	4.8 10.8 45	3.7 9.3 40
Threshold Diagonal (D)	3.471 11.17 38.64	4.2 13 40	3 9.7 35	3.471 11.17 38.64	3.471 11.17 38.64	3 10.5 35	4.1 11.9 44.37
Threshold Vertical (V)	6.148 23.08 78.39	5.5 22 64.95	7.1 25 80	5.1 21.8 75	7.3 25 84	6.148 23.08 78.39	6.148 23.08 78.39
MSE	24.8720	23.2878	24.88.39	25.0695	32.0096	24.8869	24.6607
RMSE	4.9872	4.8257	4.9884	5.0069	5.6577	4.9887	4.9660
PSNR	34.1737	34.4595	34.1716	34.1394	33.0780	34.1711	34.2107

RE – Retained Energy
NOZ- Number of Zeros

For analysis on image 1 for a compression ratio 5:1, D8 gives the best result with the lowest RMSE and PSNR. So, the explanation will be on this result. The main objective of this part is to analyze the changes of threshold values on the RMSE and PSNR. For example, if the threshold values for horizontal are fixed, so for we can observe any changes on error by varying the threshold values for diagonal and vertical. Firstly, we define threshold values randomly as showed in column 1. After that, we fixed the value for horizontal, and increase the threshold value for diagonal but decrease for vertical. Next, we fixed again the threshold value for horizontal but now increase the threshold value for vertical and decrease for diagonal. After fixing the threshold values for horizontal, we do the same thing for diagonal and vertical. So, the lowest error and highest PSNR is a blued-column. So, it can be observed that, for fix threshold values of horizontal, we can reduce more coefficients diagonally or reduce few coefficients vertically.

Table 14: Result summary for Image 1 for CR 5:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	99.69	80	33.9531	5.8269	32.8220	+ D , -V
D4	99.82	80	38.7520	6.2251	32.2479	+ D , -V
D6	99.75	80	33.7417	5.8088	32.8491	Fix D
D8	99.88	80	23.2878	4.8257	34.4595	+D, -V
D10	99.79	80	25.5059	5.0503	34.0644	Fix D

Comment:

+D = eliminate more coefficients diagonally

-V = eliminate few coefficients vertically

Fix D = fix value of threshold

From Table 14, the result agrees that for image 1 (“Lena”), for a fix threshold value horizontally, we could eliminate more coefficients diagonally and eliminate few coefficients vertically in order to get the lowest error and highest PSNR.

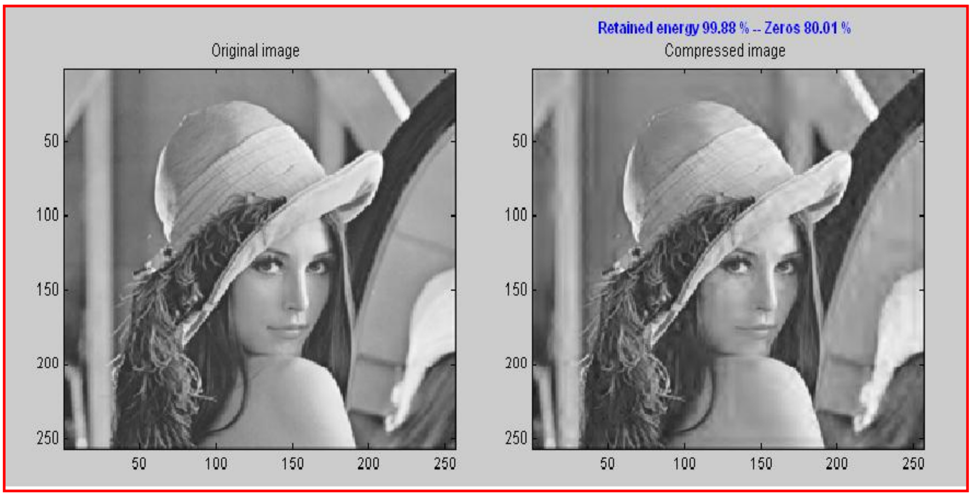


Figure 57: The best result for Image 1 compression using D8 for CR 5:1 with RMSE=4.8257 and PSNR=34.4595

Table 15: Analysis at CR=10:1, using D8 at level 3

RE(%)	99.42	99.43	99.36	99.42	99.41	99.40	99.42
NOZ(%)	90	90	90	90	90	90	90
Threshold Horizontal (H)	12.43 31.4 132.6	12.43 31.4 132.6	12.43 31.4 132.6	13.5 34 145	12 30 130	15.2 37 150	12.2 30 130
Threshold Diagonal (D)	3.471 17.74 58.68	4 19 63	3.1 16.5 56	12.43 31.4 132.6	12.43 31.4 132.6	3 17 55	3.58 17.9 60
Threshold Vertical (V)	15.61 64.85 163.5	13.4 62 160	18 70 190	14.5 63 160	16.4 67 170	15.61 64.85 163.5	15.61 64.85 163.5
MSE	59.2933	54.2186	59.2958	52.9483	58.3097	59.1151	58.3097
RMSE	7.7002	7.3633	7.7004	7.2766	7.6361	7.6886	7.6361
PSNR	30.4007	30.7893	30.4006	30.8923	30.4734	30.4138	30.4734

RE – Retained Energy
NOZ- Number of Zeros

For image 1 compression at compression ratio 10:1, again D8 produces the best result compared to other filters. As discussed for results in Table 13, the same thing will be analyzed here. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the second case where the threshold values for diagonal are fixed. The lowest RMSE was observed when we reduced the elimination of coefficients vertically and increase horizontally. This observation also agrees with the result in column 1 and 2 for fix threshold values horizontally where as we reduced the percentage of zeros, the error is lower.

Table 16: Result summary for Image 1 for CR 10:1

	RE (%)	**NOZ (%)**	**MSE**	**RMSE**	**PSNR**	**Comment**
Haar	99.23	90	60.9688	7.8082	30.2797	Fix V
D4	99.27	90	69.5760	8.3412	27.7062	-V, +H
D6	99.31	90	74.5991	8.6371	29.4035	-V, +H
D8	99.42	90	52.9483	7.2766	30.8923	-V, +H
D10	99.45	90	54.0925	7.3548	30.7994	Fix V

Comment:

+H = eliminate more coefficients diagonally

-V = eliminate few coefficients vertically

Fix V = fix value of threshold

From Table 16, the result agrees that for image 1 (“Lena”), for a fix threshold value diagonally, we could eliminate more coefficients horizontally and eliminate few coefficients vertically in order to get the lowest error and highest PSNR.

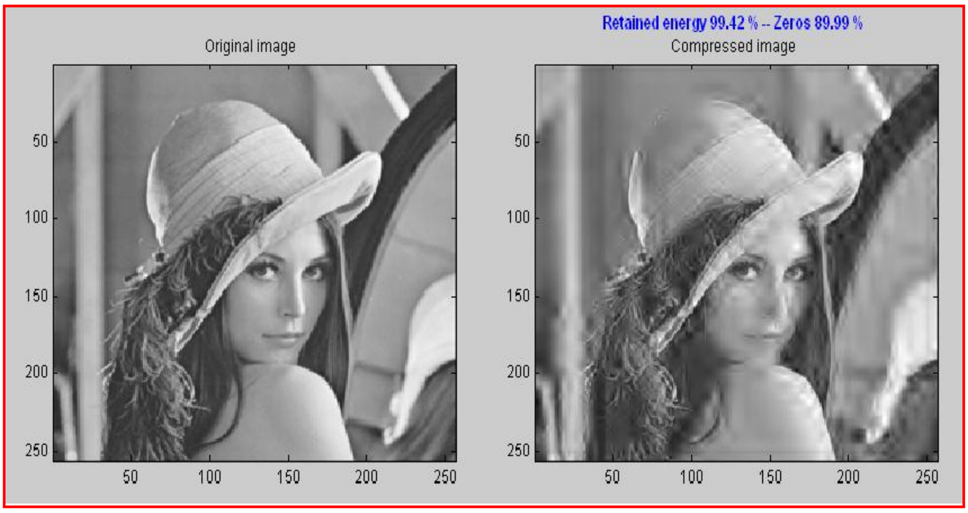


Figure 58: The best result for Image 1 compression using D8 for CR 10:1 with RMSE=7.2766 and PSNR=30.8923

Table 17: Analysis at CR=20:1, using D8 at level 3

RE(%)	99.18	99.24	99.12	99.22	99.07	99.16	99.14
NOZ(%)	95	95	95	95	95	95	95
Threshold Horizontal (H)	19.45 51.45 146.3	19.45 51.45 146.3	19.45 51.45 146.3	26 64 180	16 56 190	26 64 180	16 56 190
Threshold Diagonal (D)	11.34 34.82 121.7	18 50 160	8.7 30 95	11.34 34.82 121.7	11.34 34.82 121.7	9 30 100	13 39 155
Threshold Vertical (V)	23.17 64.85 226.2	19 60 195	28 75 260	20 60 195	25 70.5 265	23.17 64.85 226.2	23.17 64.85 226.2
MSE	86.5032	79.4682	75.8029	77.8058	88.8245	74.2175	88.8117
RMSE	9.3007	8.9145	8.7065	8.8208	9.4247	8.6150	9.4240
PSNR	28.7605	29.1289	29.3339	29.2207	28.6455	29.4257	28.6461

RE – Retained Energy
NOZ- Number of Zeros

For image 1 compression at compression ratio 20:1, again D8 produces the best result compared to other filters. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the third case where the threshold values for vertical are fixed. The lowest RMSE was observed when we reduced the percentage of zeros diagonally and increase horizontally. This observation also agrees with the result in column 3 (-D), and 4 (+H) where the observed RMSE is lower than RMSE in first column.

Table 18: Result summary for Image 1 for CR 20:1

	RE (%)	**NOZ (%)**	**MSE**	**RMSE**	**PSNR**	**Comment**
Haar	98.77	95	91.4219	9.5615	28.5203	+D, -H
D4	98.86	95	92.9110	9.6390	28.4501	+H
D6	99.06	95	80.8547	9.9919	29.0537	+H
D8	99.16	95	74.2175	8.6150	29.4257	+H , -D
D10	98.98	95	86.4423	9.2974	28.7635	-D

Comment:

+H = eliminate more coefficients diagonally

-D = eliminate few coefficients vertically

From Table 18, the result agrees that for image 1 (“Lena”), for a fix threshold value vertically, we could eliminate more coefficients horizontally and eliminate few coefficients diagonally in order to get the lowest error and highest PSNR. But, there is one “error” as highlighted in red color where the filter is Haar. Actually the error happens because all the RMSE and PSNR observed from analysis are all the same except for the “error” where the error is a bit lower than the rest but overall it did not really affect the results.

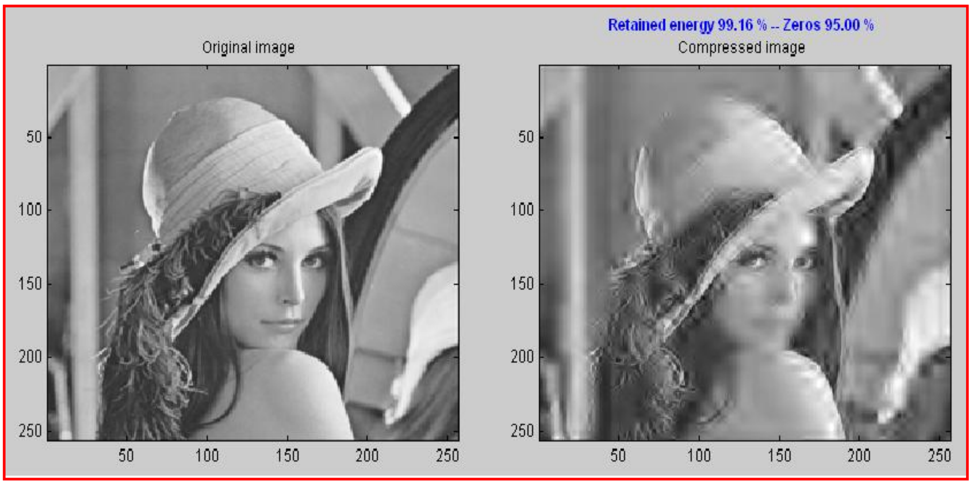


Figure 59: The best result for Image 1 compression using D8 for CR 20:1 with RMSE=8.6150 and PSNR=29.4257

4.3.2 Image 2 (“Cameraman”)

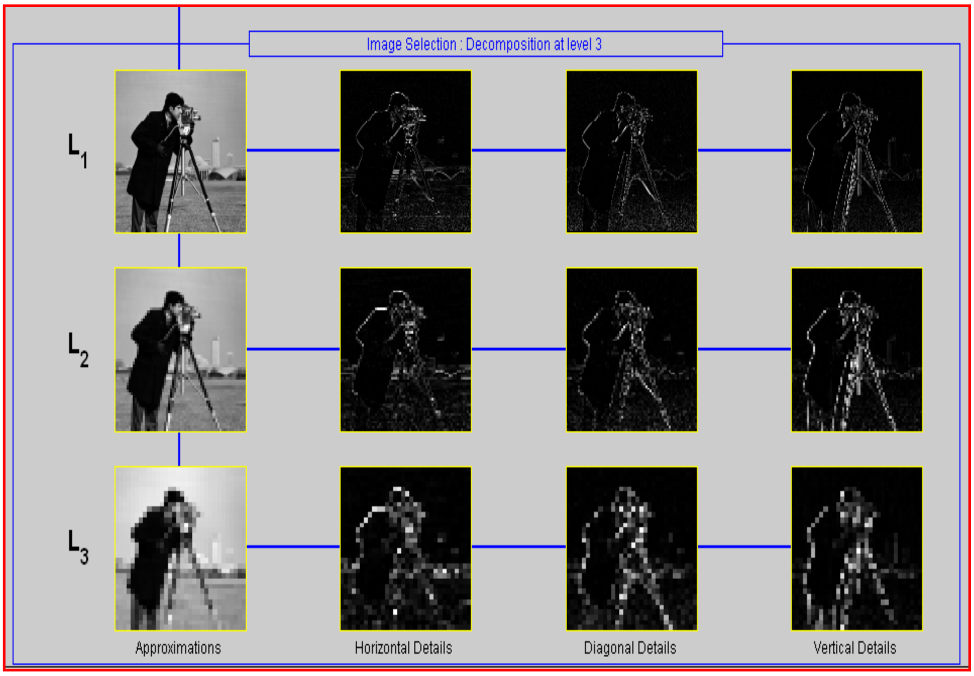


Figure 60a: Image 2 decomposition at level 3 using Haar

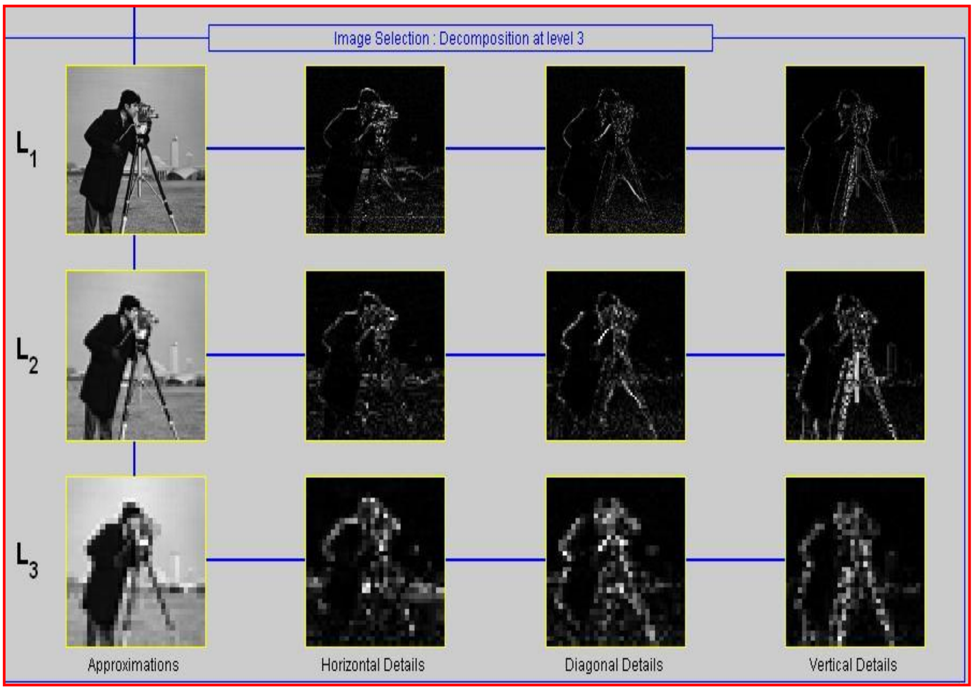


Figure 60b: Image 2 decomposition at level 3 using D4

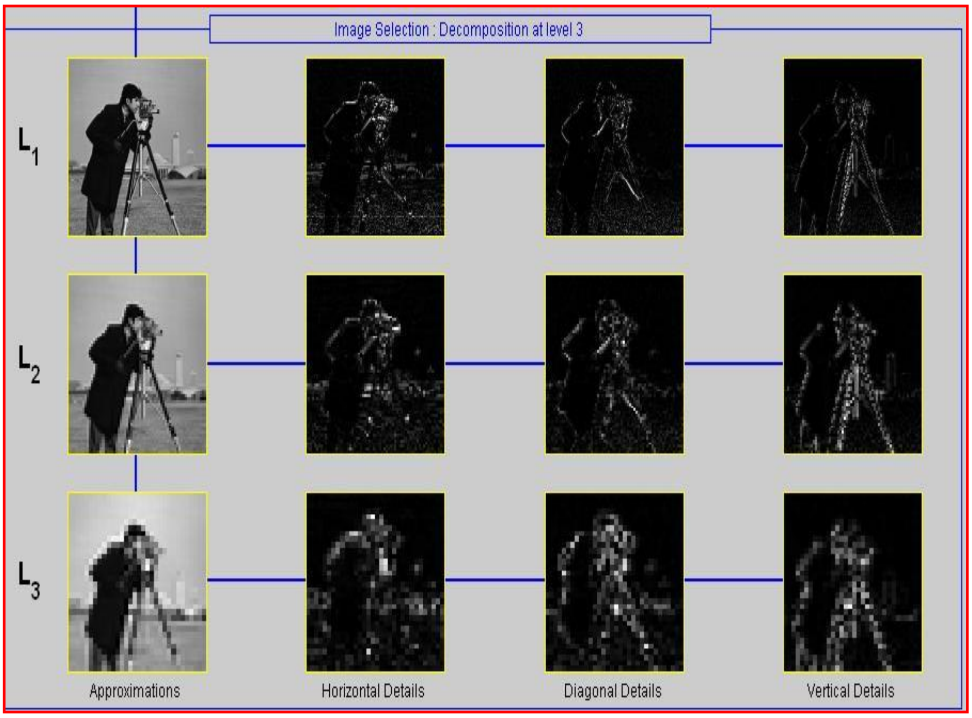


Figure 60c: Image 2 decomposition at level 3 using D6

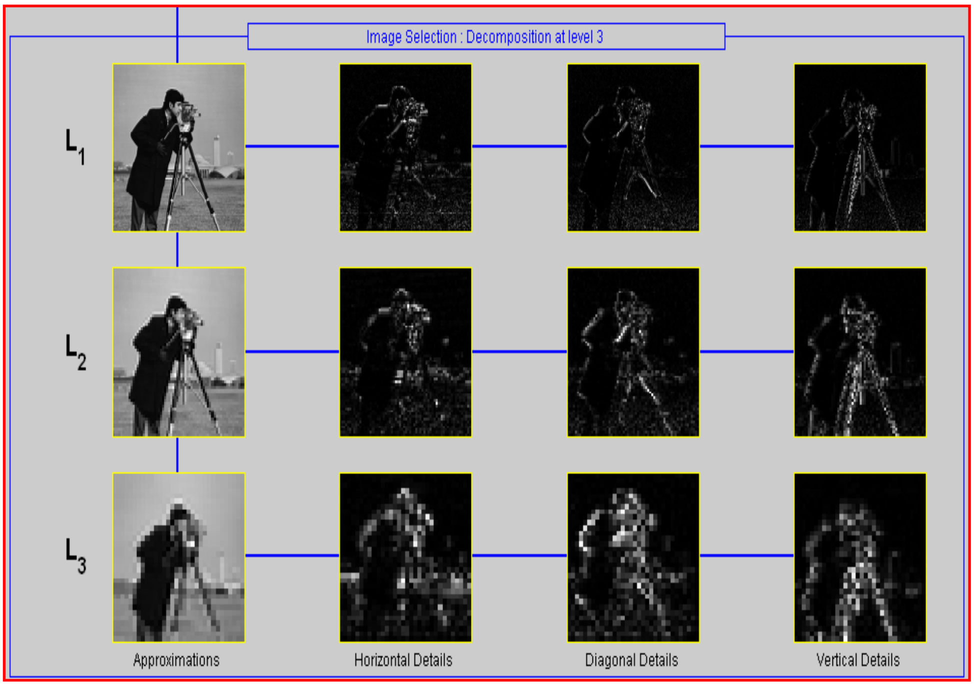


Figure 60d: Image 2 decomposition at level 3 using D8

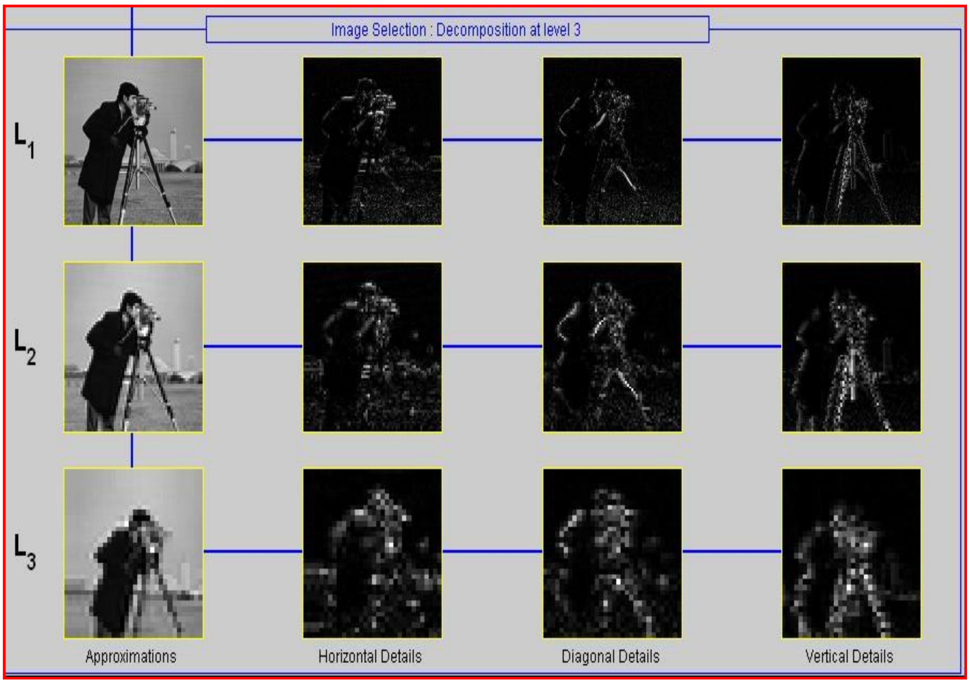


Figure 60e: Image 2 decomposition at level 3 using D10

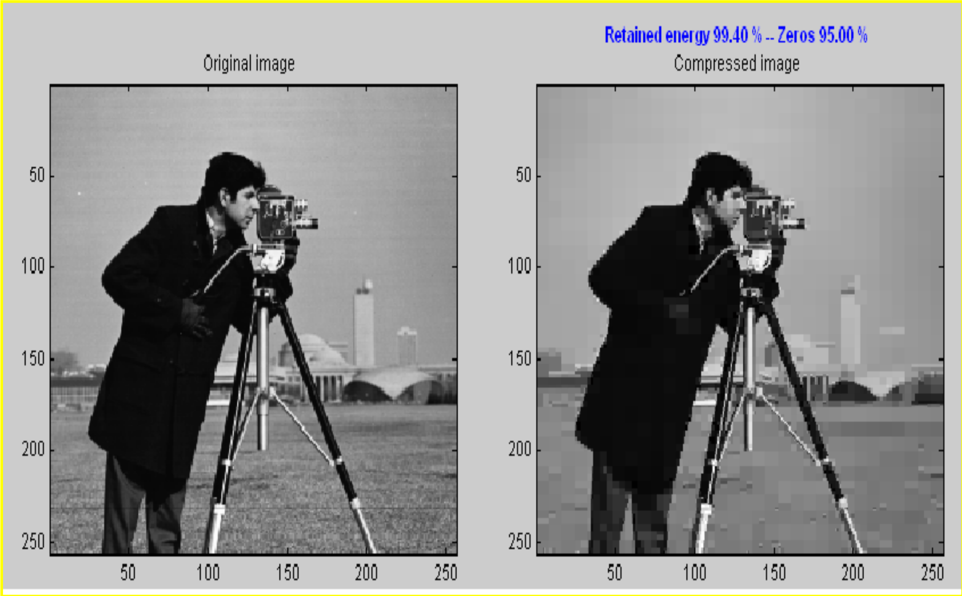


Figure 61a: Image 2 compression using Haar

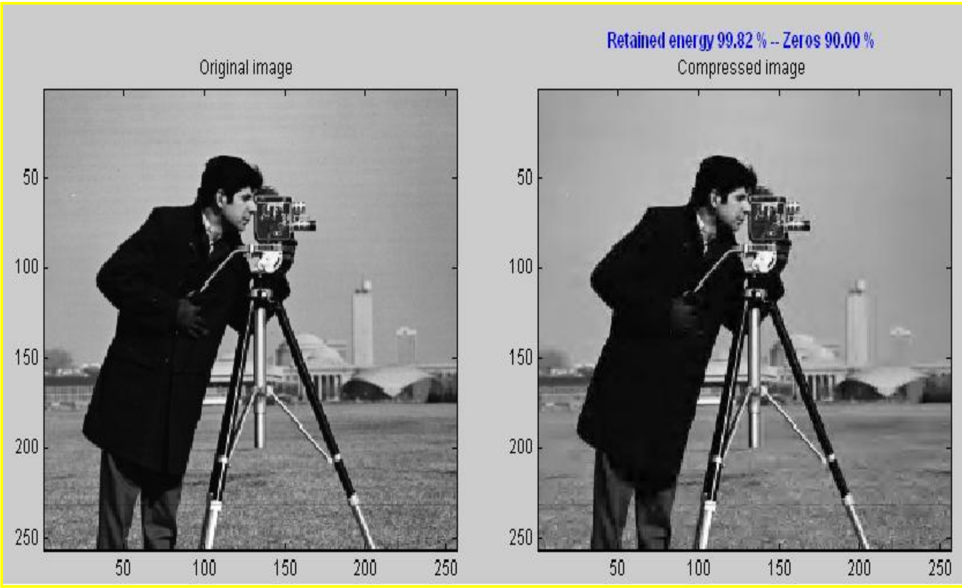


Figure 61b: Image 2 compression using D4

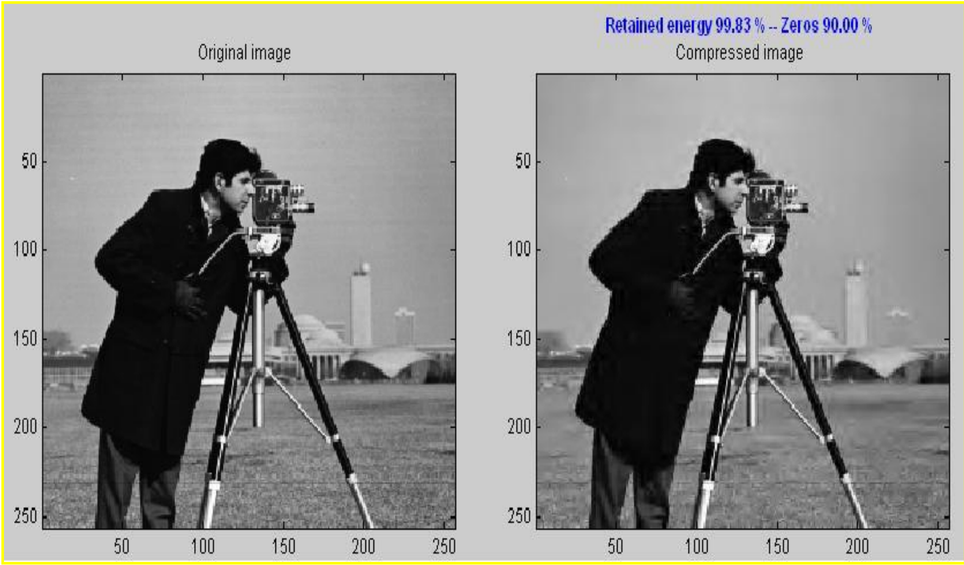


Figure 61c: Image 2 compression using D6

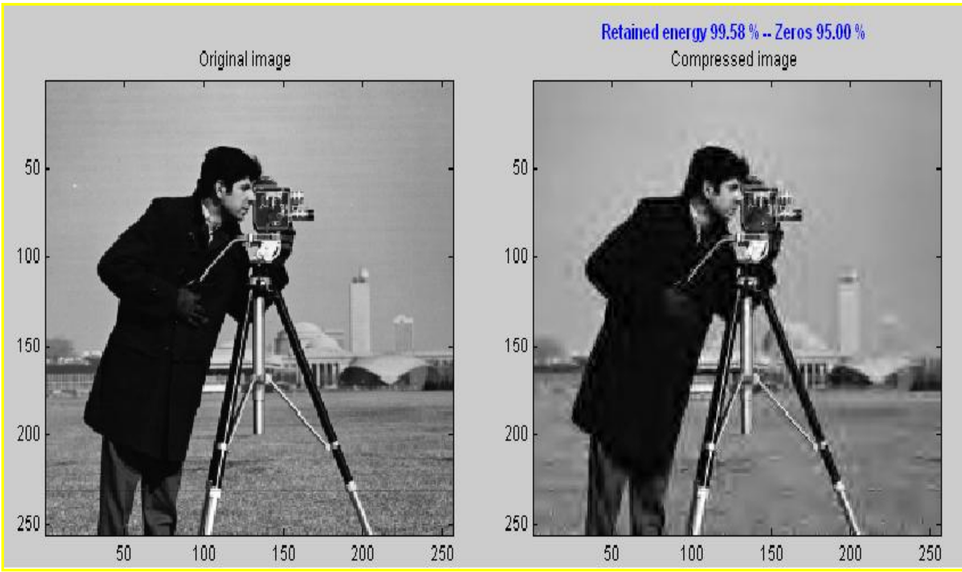


Figure 61d: Image 2 compression using D8

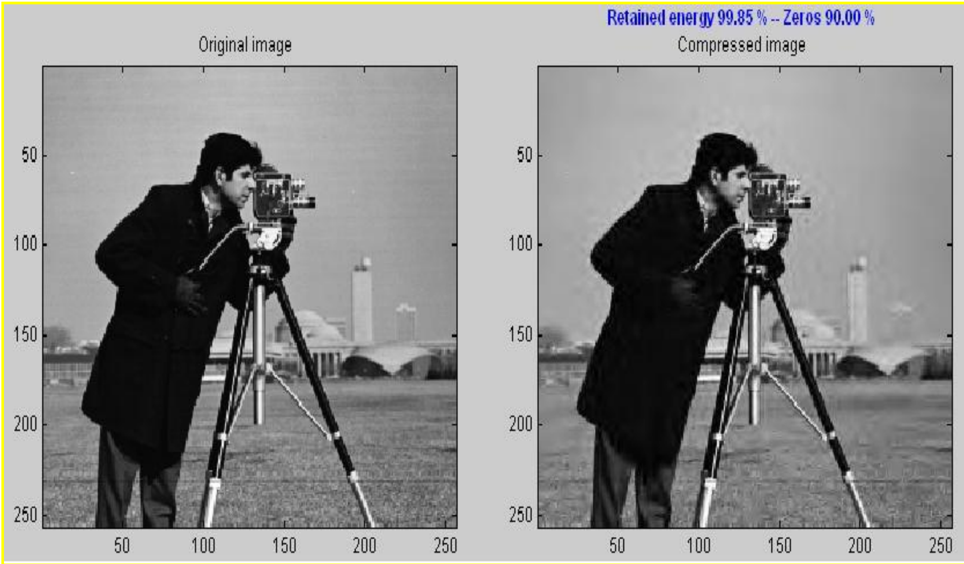


Figure 61e: Image 2 compression using D10

Figure 60 shows the wavelet decomposition at level three using 5 differen wavelet filters. Figure 61 shows the original and compressed images using Haar, D4, D6, D8 and D10. The purpose of showing these figures is to give a clear observation that Haar is different with other Daubechies filters although Haar is one of the Daubechies type. For wavelet decomposition, from an original image, wavelet decomposes it into 3 parts (horizontal, vertical and diagonal). Level 3 decomposition means wavelet decomposes an image three times.

Table 19: Analysis at CR=5:1, using D8 at level 3

RE(%)	99.93	99.93	99.92	99.92	99.92	99.92	99.93
NOZ(%)	80	80	80	80	80	80	80
Threshold Horizontal (H)	7.958 22.33 44	7.958 22.33 44	7.958 22.33 44	8.5 24 48	6.8 20 40	8.6 26 52	6.5 20 40
Threshold Diagonal (D)	6.099 12.06 28.78	7.5 15 35	5.2 10.6 22	6.099 12.06 28.78	6.099 12.06 28.78	5.5 11 25	7.3 15.1 35
Threshold Vertical (V)	12.69 17.82 47.27	10 14 46	15 24 55	12 15.5 46	14.9 23 54	12.69 17.82 47.27	12.69 17.82 47.27
MSE	23.2536	26.5135	26.0692	25.3571	28.2069	25.5960	26.0901
RMSE	4.8222	5.1491	5.1058	5.0356	5.3110	5.0593	5.1078
PSNR	34.4659	33.8961	33.9695	34.0898	33.6273	34.0491	33.9661

RE – Retained Energy
NOZ- Number of Zeros

For Image 2 at compression ratio 5:1, the lowest MSE observed is 23.2536 with PSNR 34.4659. The blued-column is the best result but unfortunately it lies under the fix value for all (horizontal, diagonal and vertical). Because of that, there is no exact observation could be made from this result. In order to make it, further analysis need to be carried out such as more threshold values analysis to make sure the lowest error does not lies in the fix column.

Table 20: Result summary for Image 2 for CR 5:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	99.88	80	25.4531	5.6181	33.1391	-
D4	99.86	80	38.3790	6.1951	32.2899	-
D6	99.84	80	38.3232	6.1906	32.2962	-
D8	99.93	80	23.2536	4.8222	34.4659	-
D10	99.91	80	23.9538	4.8943	34.3371	-

From Table 20, D8 is the best filter to decompose Image 2 for a compression ration 5:1. But, there is no comment could be made because of the problem mentioned above.

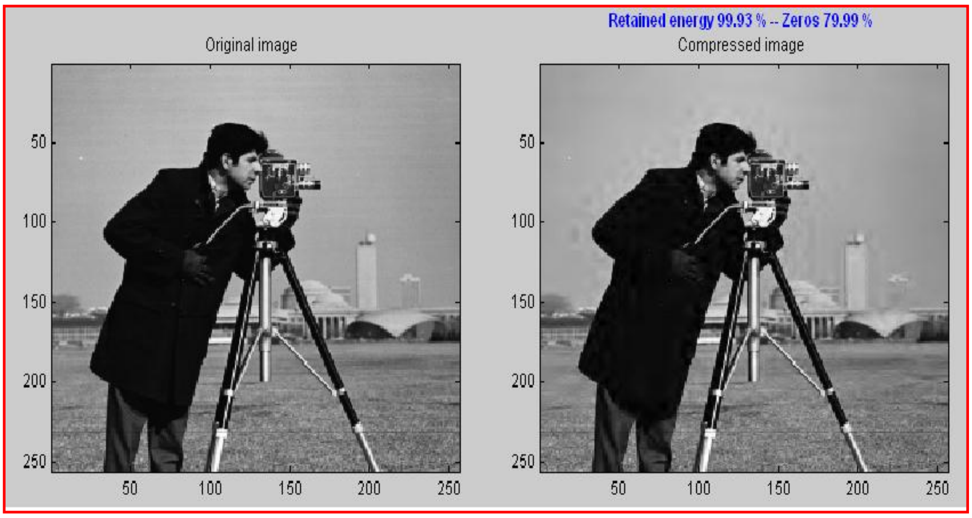


Figure 62: The best result for Image 2 compression using D8 for CR 5:1 with RMSE=4.8222 and PSNR=34.4659

Table 21: Analysis at CR=10:1, using D8 at level 3

RE(%)	99.69	99.69	99.66	99.70	99.67	99.68	99.69
NOZ(%)	90	90	90	90	90	90	90
Threshold Horizontal (H)	16.64 49.14 103.1	16.64 49.14 103.1	16.64 49.14 103.1	18 52 111	15 48 94	18 52 111	16 48 98
Threshold Diagonal (D)	14.97 21.33 69.29	16 24 75	13.3 19 63	14.97 21.33 69.29	14.97 21.33 69.29	13.35 20 63	15.75 22 82
Threshold Vertical (V)	26.54 53.46 114	24 50 111	30 60 130	23.6 50 100	30 60 130	14.97 21.33 69.29	14.97 21.33 69.29
MSE	52.8010	52.9130	55.6678	50.222	55.8057	54.0880	57.1759
RMSE	7.2664	7.2741	7.4611	7.0867	7.4703	7.3545	7.5615
PSNR	30.9044	30.8952	30.6748	31.1219	30.6640	30.7998	30.5587

RE – Retained Energy
NOZ- Number of Zeros

For Image 2 compression at compression ratio 10:1, again D8 produces the best result compared to other filters. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the second case where the threshold values for vertical are fixed. The lowest RMSE was observed when we reduced the percentage of zeros vertically (-V) and increase horizontally (+H). This observation also agrees with the result in column 2 (-V), and 6 (+H) where the observed RMSE is lower than RMSE in the first column.

Table 22: Result summary for Image 2 for CR 10:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	99.52	90	55.1719	7.4278	30.7136	-V
D4	99.56	90	76.1155	8.7244	29.3161	Fix V
D6	99.57	90	67.5980	7.8484	30.2351	Fix V
D8	99.70	90	50.222	7.0867	31.1219	+H , -V
D10	99.67	90	58.4623	7.6461	30.4620	Fix D

Comment:

+H = eliminate more coefficients diagonally

-V = eliminate few coefficients vertically

Fix V = fix value of threshold

From Table 22, the result agrees that for Image 2 (“Cameraman”), for a fix threshold value diagonally, we could eliminate more coefficients horizontally and eliminate few coefficients vertically in order to get the lowest error and highest PSNR.

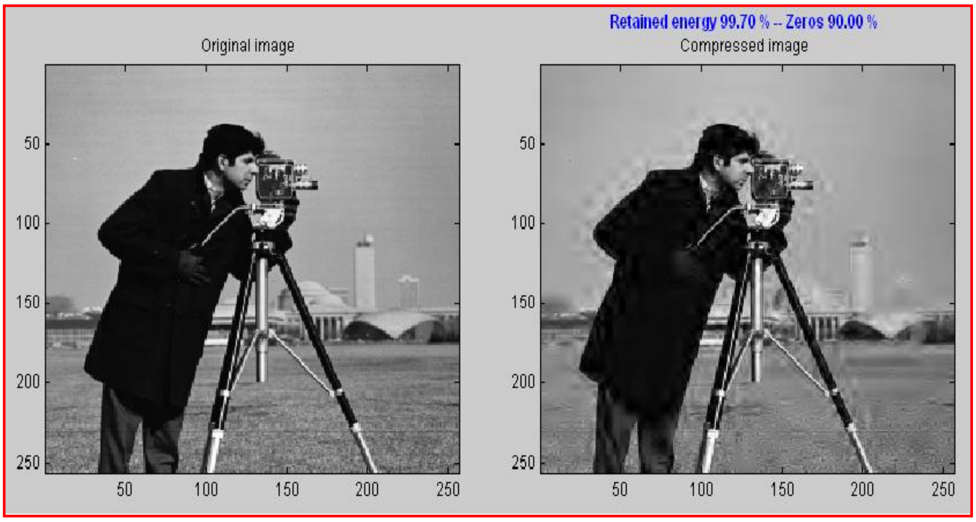


Figure 63: The best result for Image 2 compression using D8 for CR 10:1 with RMSE=7.0867 and PSNR=31.1219

Table 23: Analysis at CR=20:1, using D8 at level 3

RE(%)	99.22	99.24	99.19	99.24	99.20	99.22	99.22
NOZ(%)	95	95	95	95	95	95	95
Threshold Horizontal (H)	31.11 72.96 151.7	31.11 72.96 151.7	31.11 72.96 151.7	32.5 75 160	30 71 148	32.5 75 160	30.5 71.5 147
Threshold Diagonal (D)	23.84 47.29 111.9	25 50 130	22 45 101.3	23.84 47.29 111.9	23.84 47.29 111.9	23 45 108	24 50 130
Threshold Vertical (V)	47.32 82.61 225.2	45.5 80 215	51.6 88 250	45.7 80 215	50 86 245	47.32 82.61 225.2	47.32 82.61 225.2
MSE	98.1774	106.347	104.597	94.3366	102.241	98.1874	98.1603
RMSE	9.9084	10.3124	10.2272	9.7127	10.1114	9.9090	9.9076
PSNR	28.2107	27.8636	27.9356	28.3840	28.0346	28.2102	28.2114

RE – Retained Energy
NOZ- Number of Zeros

For Image 2 compression at compression ratio 20:1, again D8 produces the best result compared to other filters. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the second case where the threshold values for diagonal are fixed. The lowest RMSE was observed when we reduced the percentage of zeros vertically (-V) and increase horizontally (+H). However, for –V, it did not apply for all column so for this case only +H will be taken into account.

Table 24: Result summary for Image 2 for CR 20:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	98.74	95	113.6562	10.6610	27.5749	Fix H
D4	98.84	95	108.6501	10.4235	27.7705	+H
D6	98.79	95	128.0357	11.3153	27.0575	Fix H
D8	99.24	95	94.3366	9.7127	28.3840	+H
D10	99.14	95	103.7228	10.1844	27.9721	Fix H

Comment:

+H = eliminate more coefficients diagonally

Fix H = fix value of threshold

From Table 24, the result agrees that for Image 2 (“Cameraman”), for a fix threshold value diagonally, we could eliminate more coefficients horizontally in order to get the lowest error and highest PSNR. For vertical threshold, the result did not apply to all columns and because of that, only +H will be taken into consideration.

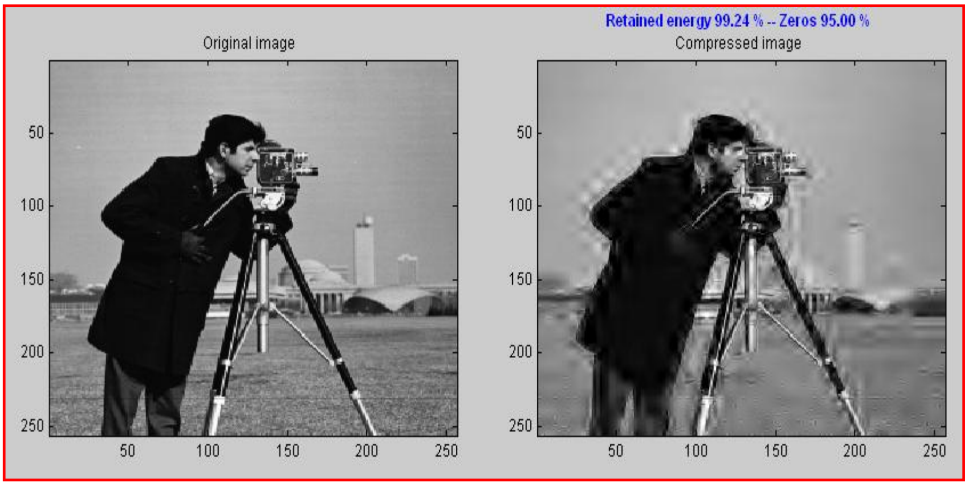


Figure 64: The best result for Image 2 compression using D8 for CR 20:1 with RMSE=9.7127 and PSNR=28.3840

4.3.3 Image 2 (“Peppers”)

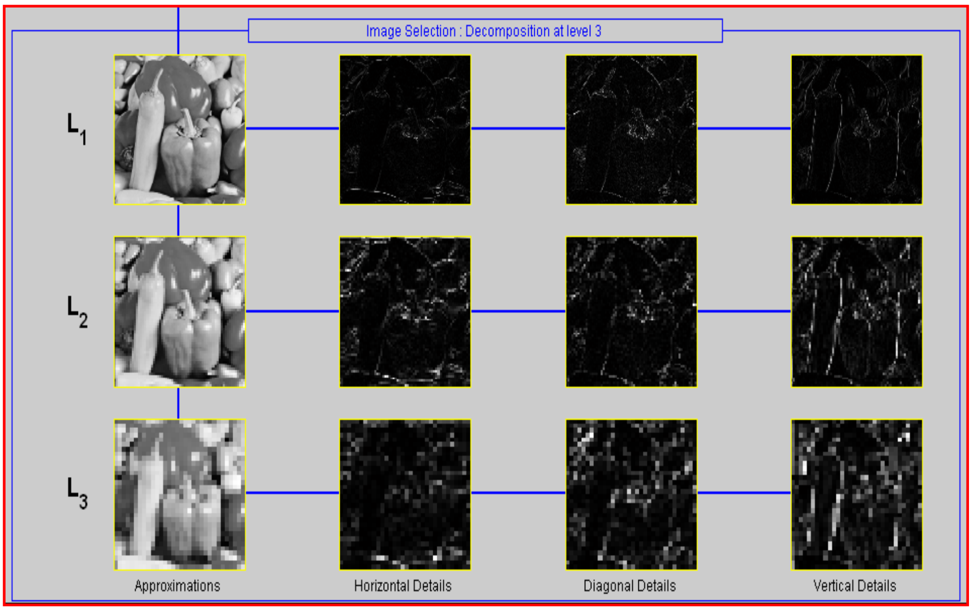


Figure 65a: Image 3 decomposition at level 3 using Haar

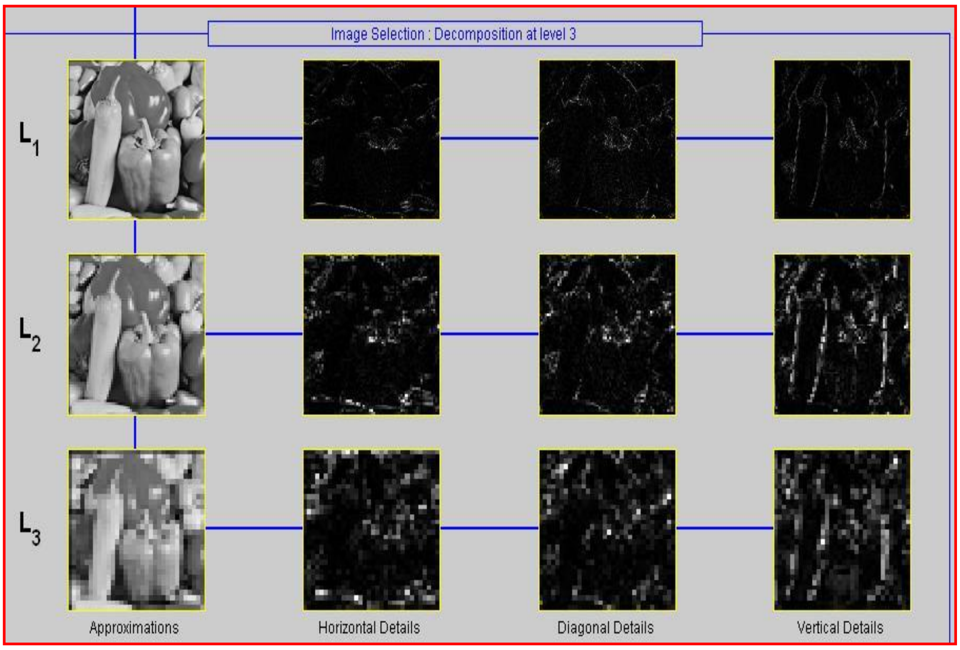


Figure 65b: Image 3 decomposition at level 3 using D4

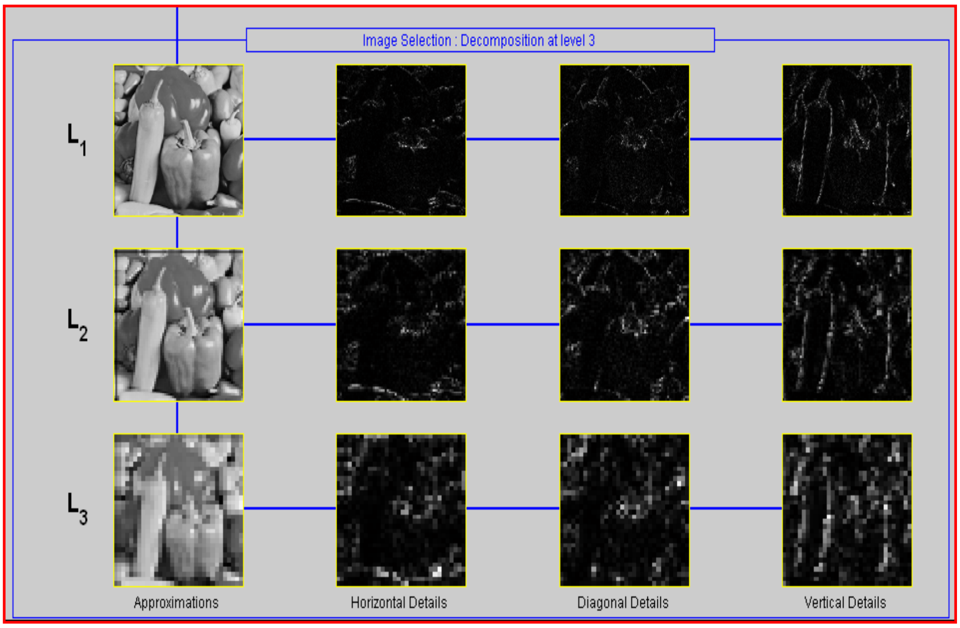


Figure 65c: Image 3 decomposition at level 3 using D6

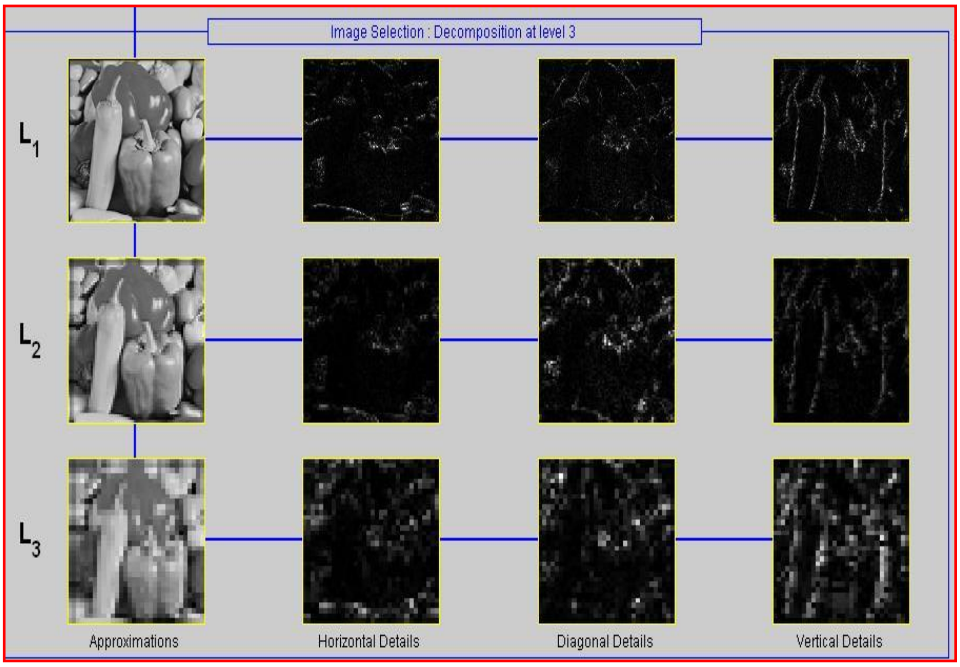


Figure 65d: Image 3 decomposition at level 3 using D8

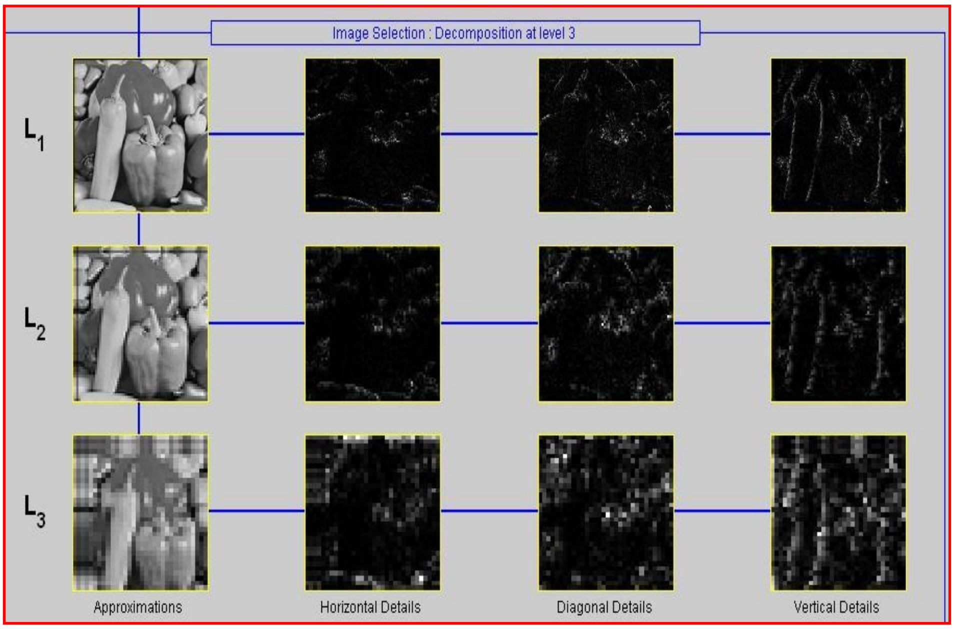


Figure 65e: Image 3 decomposition at level 3 using D10

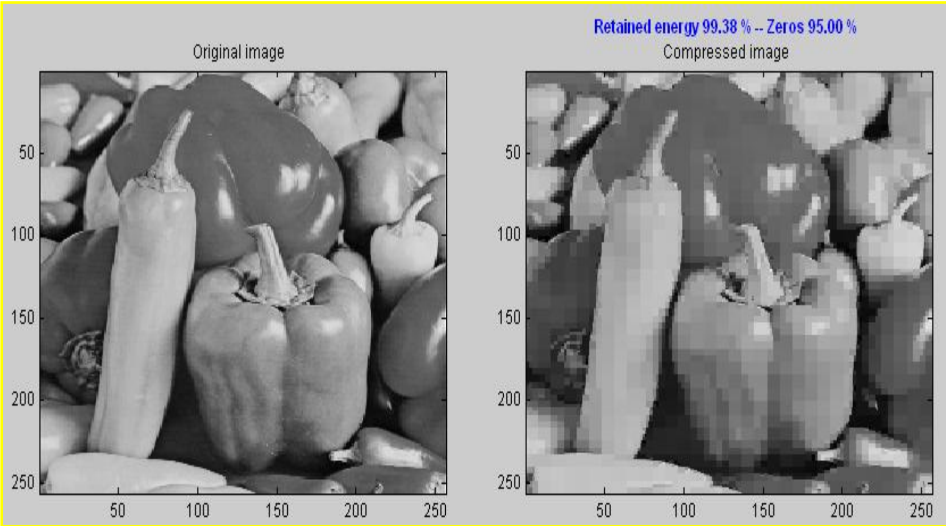


Figure 66a: Image 3 compression using Haar

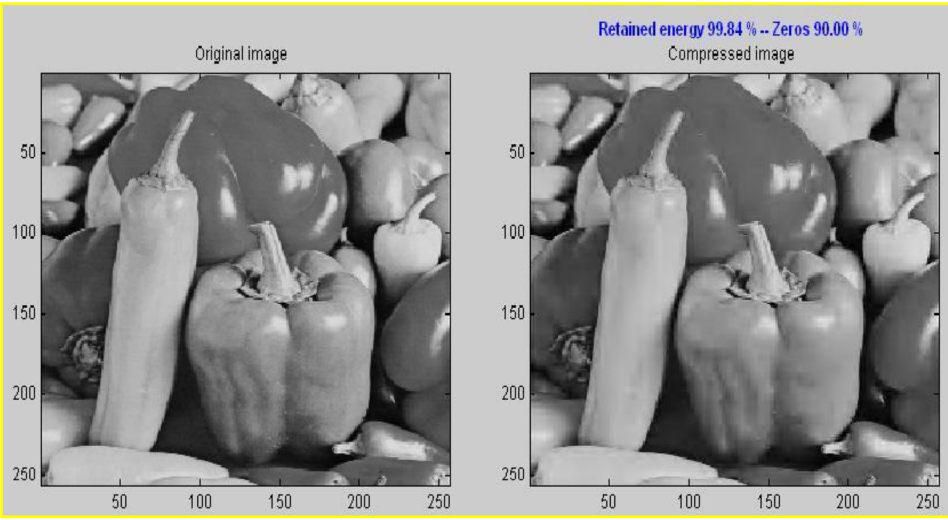


Figure 66b: Image 3 compression using D4

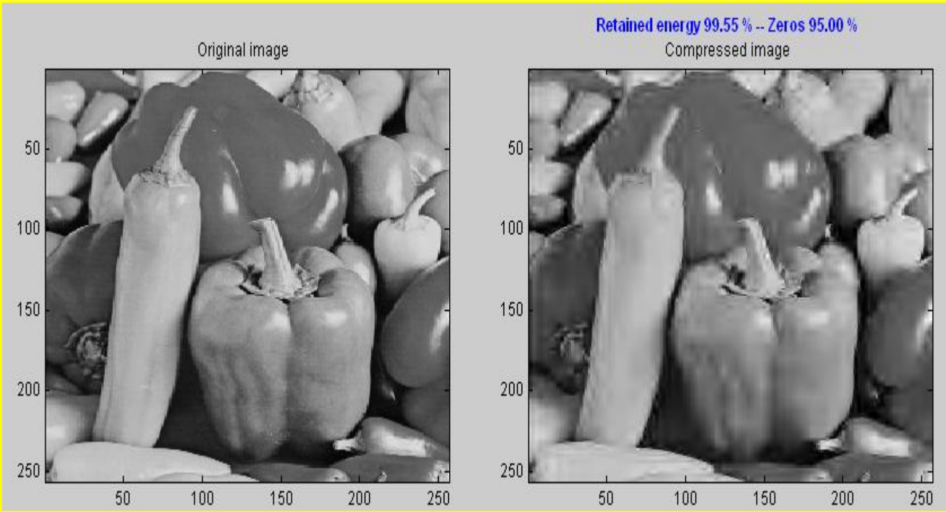


Figure 66c: Image 3 compression using D6

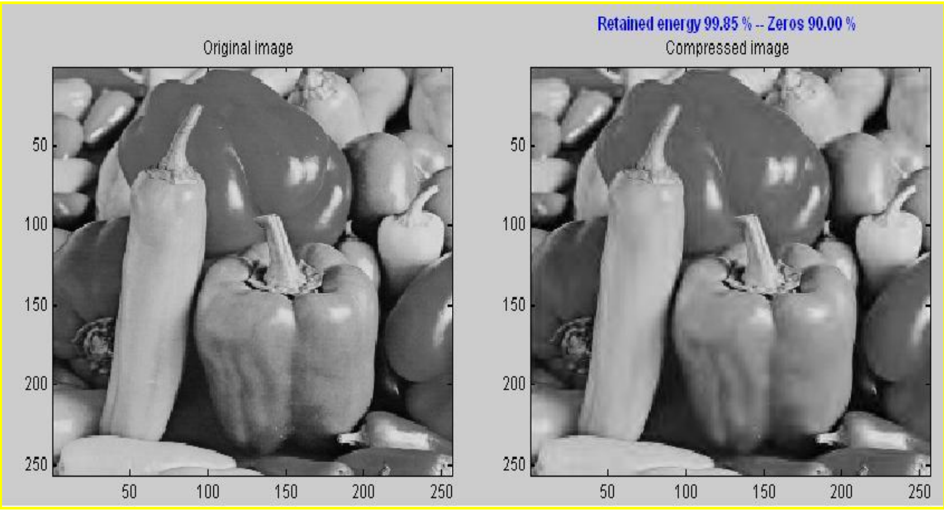


Figure 66d: Image 3 compression using D8

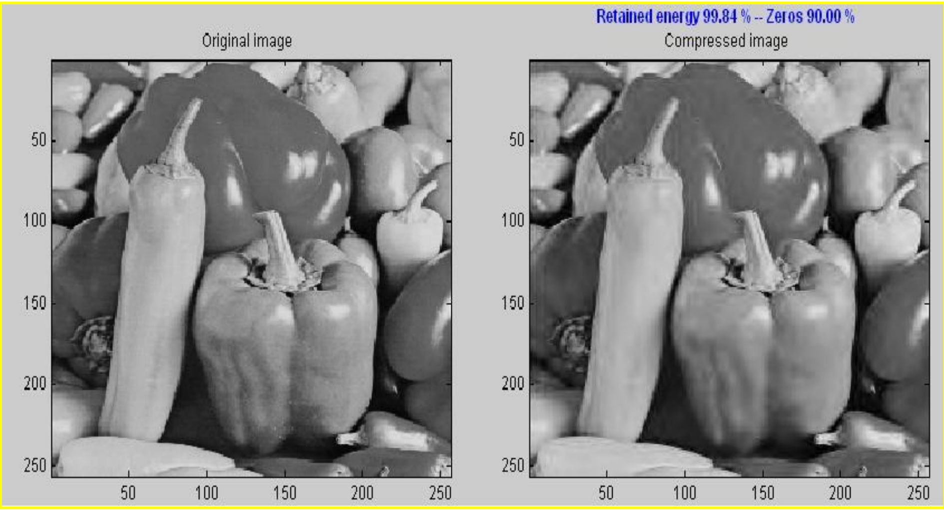


Figure 66e: Image 3 compression using D10

Figure 65 shows the wavelet decomposition at level three using 5 different wavelet filters. Figure 66 shows the original and compressed images using Haar, D4, D6, D8 and D10 respectively. The purpose of showing these figures is to give a clear observation that Haar is different with other Daubechies filters although Haar is one of the Daubechies type. For wavelet decomposition, from an original image, wavelet decomposes it into 3 parts (horizontal, vertical and diagonal). Level 3 decomposition means wavelet decomposes an image three times.

Table 25: Analysis at CR=5:1, using Haar at level 3

RE(%)	99.71	99.73	99.66	99.70	99.72	99.64	99.72
NOZ(%)	80	80	80	80	80	80	80
Threshold Horizontal (H)	12.17 24.98 96.02	12.17 24.98 96.02	12.17 24.98 96.02	13.5 28 105	10 21.5 88	17.49 32.5 125	11 22.8 94
Threshold Diagonal (D)	3.816 12.1 31.28	4.5 14 37	3.1 11 28	3.816 12.1 31.28	3.816 12.1 31.28	3.2 11.5 29	4 13 35
Threshold Vertical (V)	10.17 18.96 93.25	8 16.4 88	13 24 112	9.5 17.4 88	12.4 20 100	10.17 18.96 93.25	10.17 18.96 93.25
MSE	34.3438	34.3438	36.7656	34.3438	35.8437	46.1250	33.3750
RMSE	5.8604	5.8604	6.0635	5.8604	5.9870	6.7915	5.7771
PSNR	32.7723	32.7723	32.4764	32.7723	32.5867	31.4914	32.8966

RE – Retained Energy
NOZ- Number of Zeros

For Image 3 at compression ratio 5:1, the lowest MSE observed is 33.3750 with PSNR 32.8966. The blued-column is the best result when we fix the threshold value for vertical part. Considering the last column, the lowest error produced when increases the percentage of zeros for diagonal part (+D) as well as if we decrease the percentage of zeros for horizontal part (-H). But, the observation is not strong enough to conclude that +D and –H could be applied to other filters.

Table 26: Result summary for Image 3 for CR 5:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	99.72	80	33.3750	5.7771	32.8966	-H , +D
D4	99.71	80	49.5354	7.0381	31.1816	+D
D6	99.79	80	38.1688	6.1781	32.3137	-
D8	99.68	80	39.6277	6.2951	32.1508	Fix value
D10	99.78	80	34.1513	5.8439	32.7967	-H , +D

Comment:

+D = eliminate more coefficients diagonally

-H = eliminate few coefficients vertically

Fix value = fix value of threshold

From Table 26, Haar is the best filter to decompose Image 3 for a compression ration 5:1. From the comment, it can be said that we can reduce the error by increasing the percentage of zero diagonally. We also could try reducing the percentage of zeros horizontally.

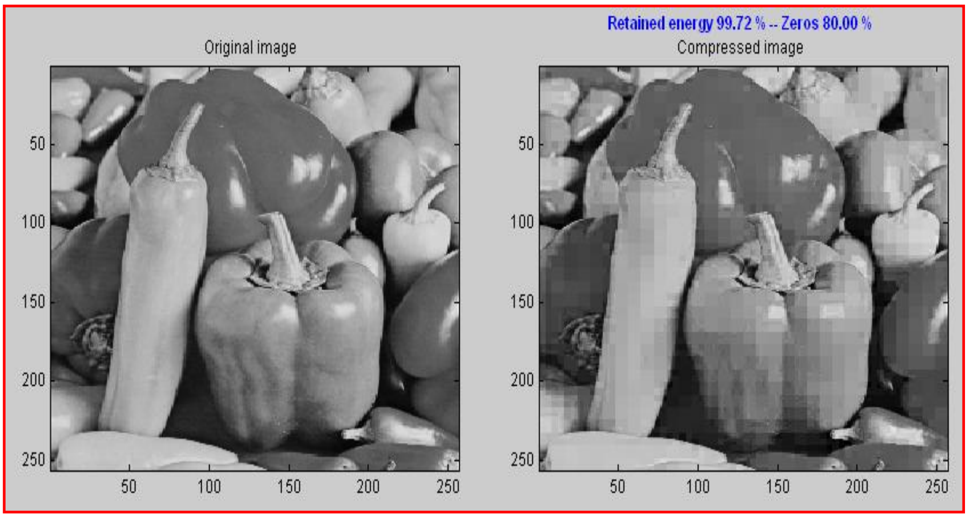


Figure 67: The best result for Image 3 compression using Haar for CR 5:1 with RMSE=5.7771 and PSNR=32.8966

Table 27: Analysis at CR=10:1, using Haar at level 3

RE(%)	99.13	99.17	99.08	99.13	99.09	99.07	99.13
NOZ(%)	90	90	90	90	90	90	90
Threshold Horizontal (H)	21.03 38.13 195.6	21.03 38.13 195.6	21.03 38.13 195.6	22 41.7 205	19 35 185	21.6 43.2 220	20 36 185
Threshold Diagonal (D)	7.208 25.01 96.36	8.5 28 100	6.8 22 90	7.208 25.01 96.36	7.208 25.01 96.36	7 23.5 90	7.5 27 125
Threshold Vertical (V)	21.47 42.29 189	18 38 175	23.5 46.5 200	20.4 41 180	23.4 46 210	21.47 42.29 189	21.47 42.29 189
MSE	67.5156	69.4063	70	67.5156	65.1875	71.4062	69.4063
RMSE	8.2168	8.3310	8.3666	8.2168	8.0739	8.4502	8.3310
PSNR	29.8368	29.7168	29.8368	29.8368	29.9892	29.5934	29.7168

RE – Retained Energy
NOZ- Number of Zeros

For Image 3 compression at compression ratio 10:1, again Haar produces the best result compared to other filters. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the second case where the threshold values for diagonal are fixed. The lowest RMSE was observed when we increased the percentage of zeros vertically (+V) and decreased horizontally (-H). This observation was supported for other columns like column 3 (+V) and column 7 (-H).

Table 28: Result summary for Image 3 for CR 10:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	99.09	90	65.1875	8.0739	29.9892	-H , +V
D4	99.03	90	82.7408	9.0962	28.9536	-
D6	99.06	90	93.0514	9.6463	28.4436	-H
D8	98.83	90	98.6180	9.9307	28.1912	-H , +V
D10	98.87	90	77.1641	8.7843	29.2566	-H, +V

Comment:

+V = eliminate more coefficients diagonally
-H = eliminate few coefficients horizontally
Fix H = fix value of threshold

From Table 28, the result agrees that for Image 3 ("Peppers"), for a fix threshold value diagonally, we could eliminate more coefficients vertically and eliminate few coefficients horizontally in order to get the lowest error and highest PSNR. All filters follow the same trend except for D4.

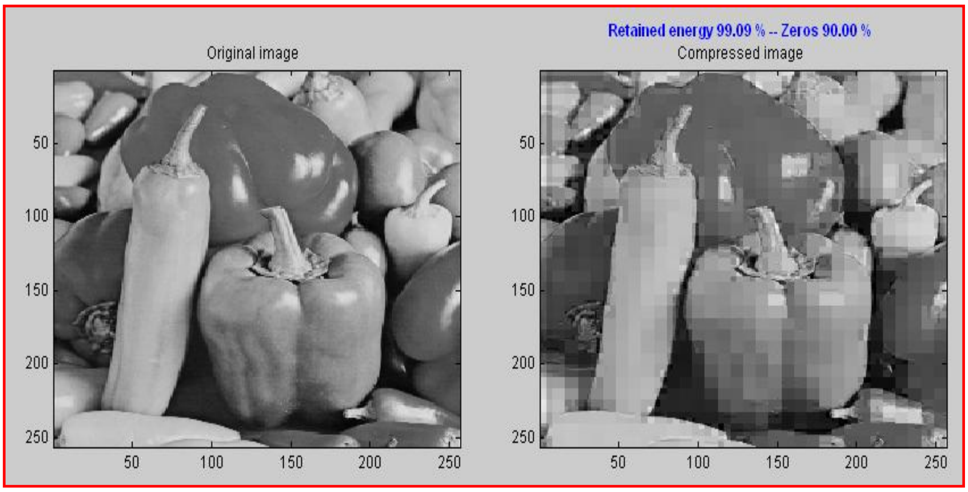


Figure 68: The best result for Image 3 compression using Haar for CR 10:1 with RMSE=8.0739 and PSNR=29.9892

Table 29: Analysis at CR=20:1, using D4 at level 3

RE(%)	98.48	98.48	98.41	98.46	98.48	98.45	98.55
NOZ(%)	95	95	95	95	95	95	95
Threshold Horizontal (H)	36.82 78.8 229.9	36.82 78.8 229.9	36.82 78.8 229.9	38.5 81 245	33 73 300	40.4 80 333	31 71 300
Threshold Diagonal (D)	15.21 32.58 109.3	17 35 125	13 30 100	15.21 32.58 109.3	15.21 32.58 109.3	12 45.2 120	15 50 138
Threshold Vertical (V)	30.29 62.58 325.5	28.5 60.5 320	34 67 355	29.6 61 320	41 120 340	30.29 62.58 325.5	30.29 62.58 325.5
MSE	112.513	121.164	111.579	112.513	117.094	117.094	103.156
RMSE	10.6072	11.0075	10.5631	10.6072	10.8210	10.8210	10.1562
PSNR	27.6188	27.2971	27.6550	27.6188	27.4455	27.4455	27.9958

RE – Retained Energy
NOZ- Number of Zeros

For Image 3 compression at compression ratio 20:1, now D4 produces the best result compared to other filters. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the second case where the threshold values for diagonal are fixed. The lowest RMSE was observed when we reduced the percentage of zeros horizontally (-H) and increase vertically (+V). Other 2 columns also satisfied by the observation where column 3 (+V) and column 7 (-H).

Table 30: Result summary for Image 3 for CR 20:1

	RE (%)	**NOZ (%)**	**MSE**	**RMSE**	**PSNR**	**Comment**
Haar	98.19	95	103.1562	10.1562	27.9958	-H
D4	98.48	95	101.3799	10.0688	28.0713	-H , +V
D6	98.24	95	109.3247	10.4558	27.7436	-H
D8	98.48	95	114.2243	10.6876	27.5532	+V
D10	98.41	95	110.1994	10.4976	27.7090	-

Comment:

+V = eliminate more coefficients diagonally

-H = eliminate few coefficients horizontally

From Table 30, the result agrees that for Image 3 ("Peppers"), for a fix threshold value diagonally, we could eliminate more coefficients vertically and eliminate few coefficients horizontally in order to get the lowest error and highest PSNR.

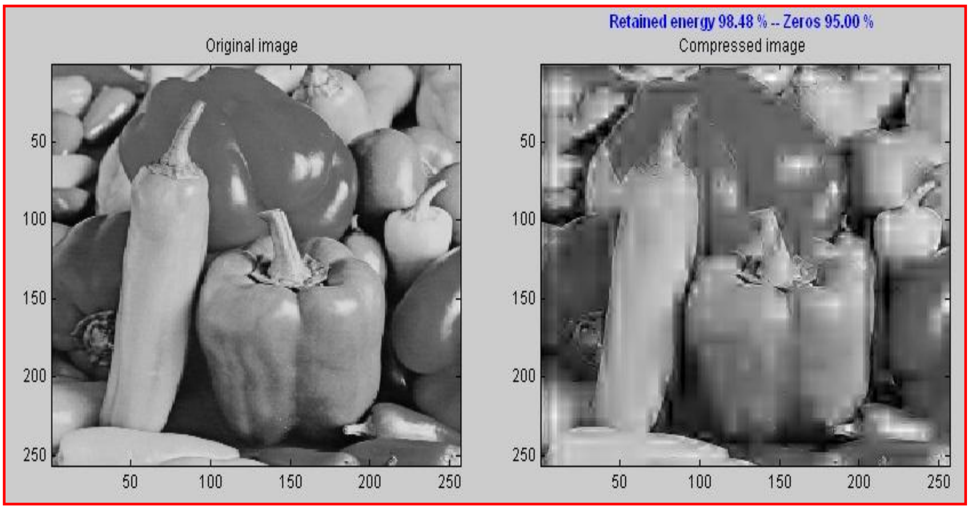


Figure 69: The best result for Image 3 compression using D4 for CR 20:1 with RMSE=10.0688 and PSNR=28.0713

4.3.4 Image 4 ("House")

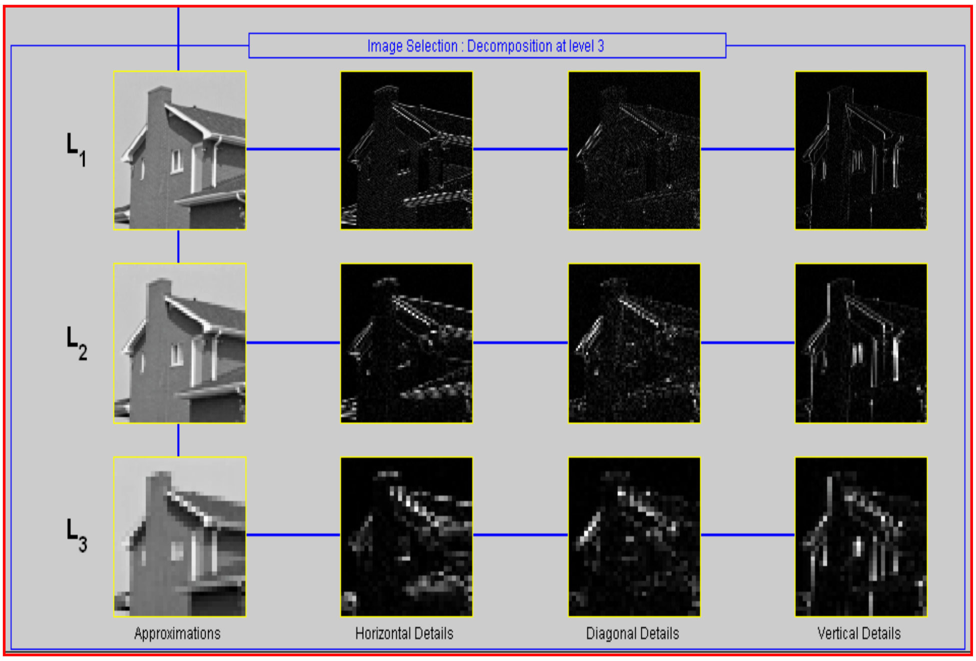


Figure 70a: Image 4 decomposition at level 3 using Haar

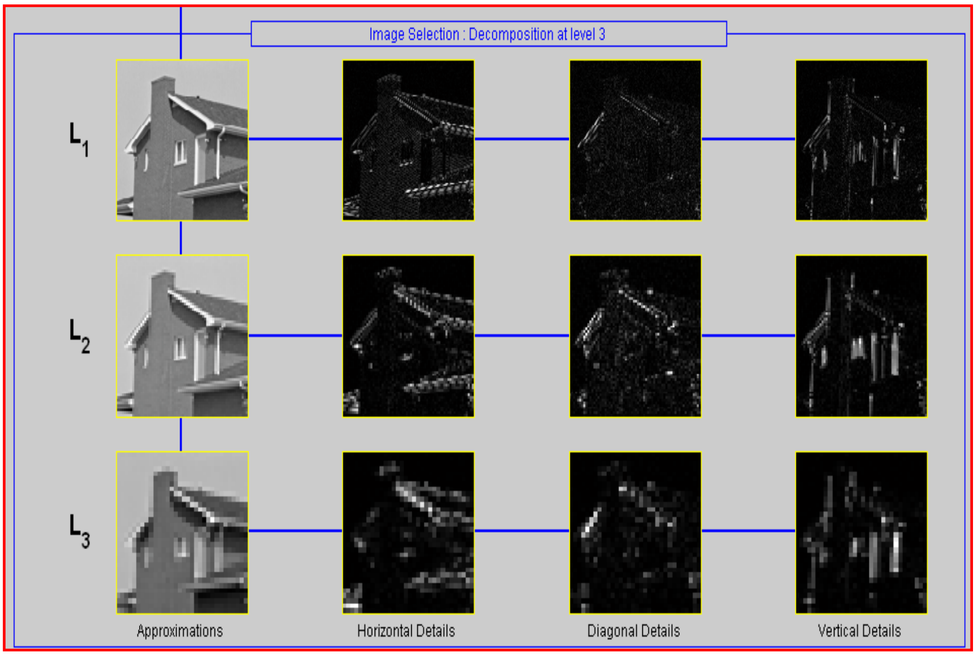


Figure 70b: Image 4 decomposition at level 3 using D4

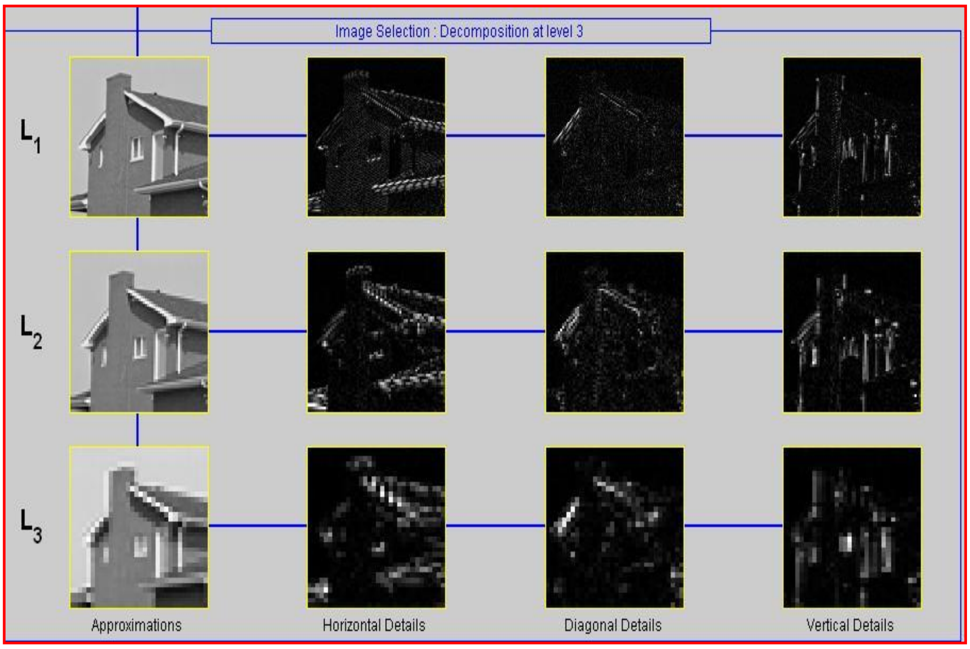


Figure 70c: Image 4 decomposition at level 3 using D6

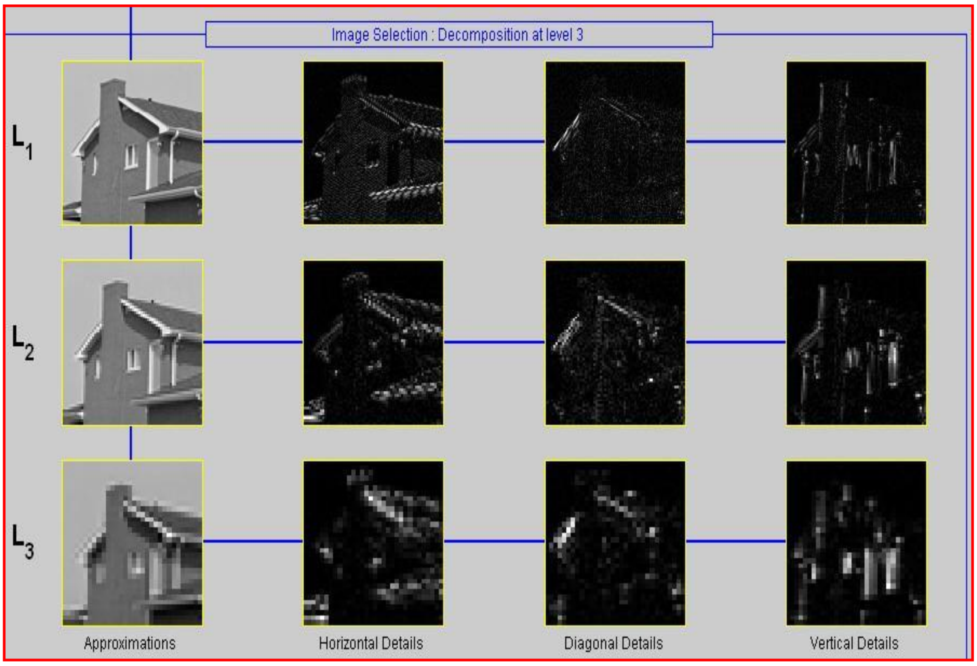


Figure 70d: Image 4 decomposition at level 3 using D8

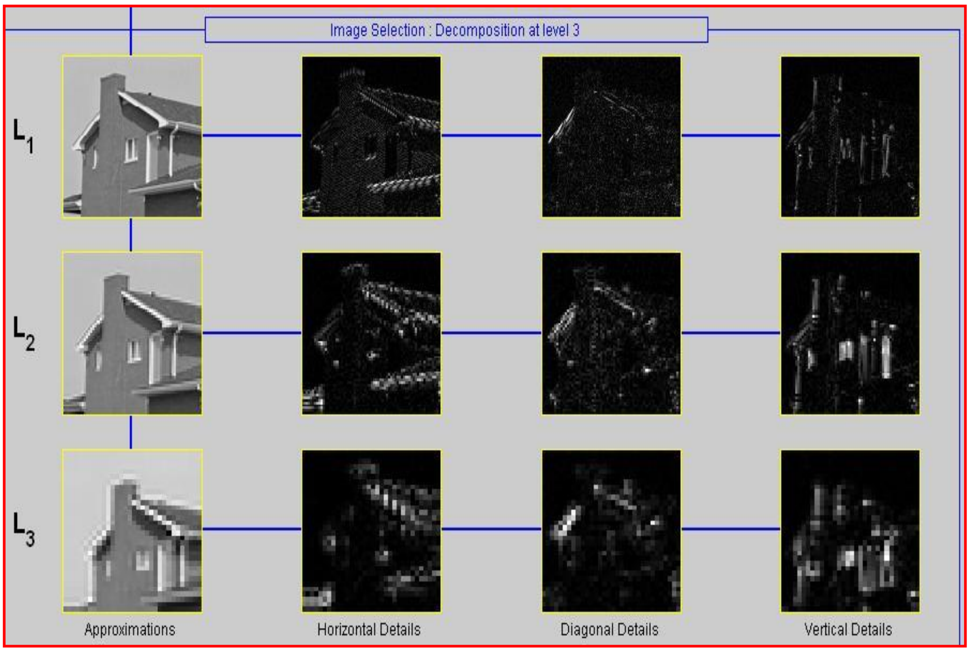


Figure 70e: Image 4 decomposition at level 3 using D10

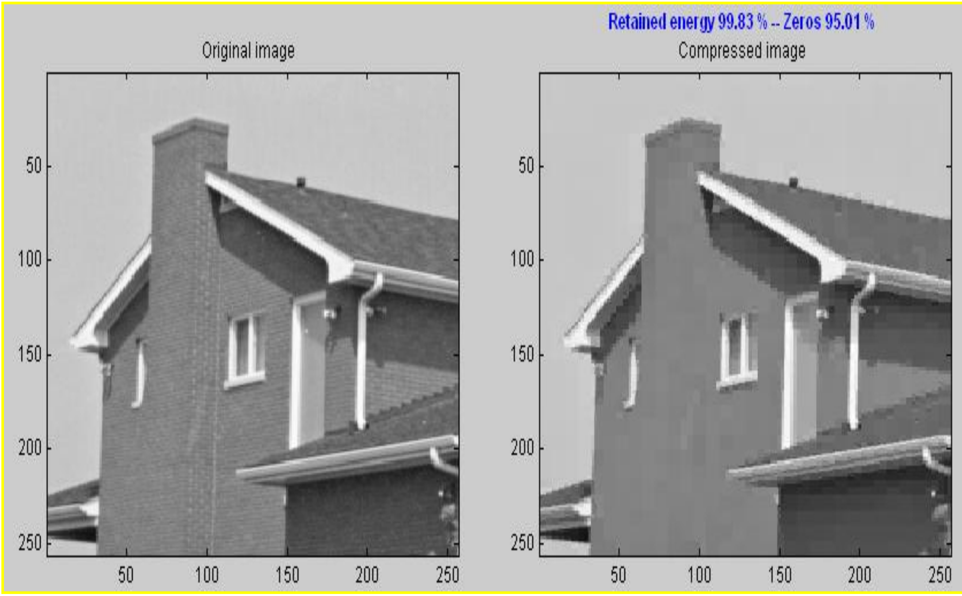


Figure 71a: Image 4 compression using Haar

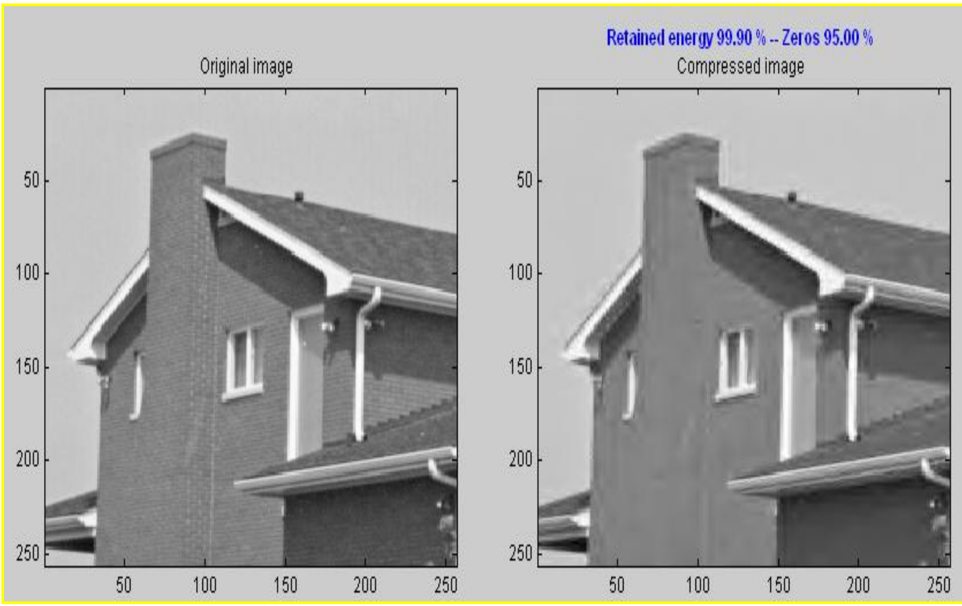


Figure 71b: Image 4 compression using D4

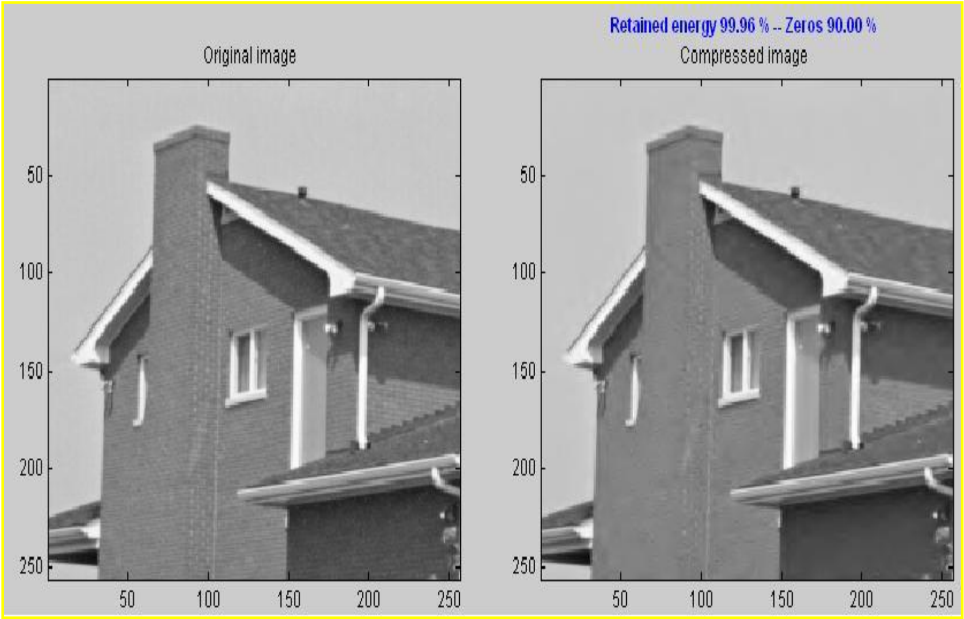


Figure 71c: Image 4 compression using D6

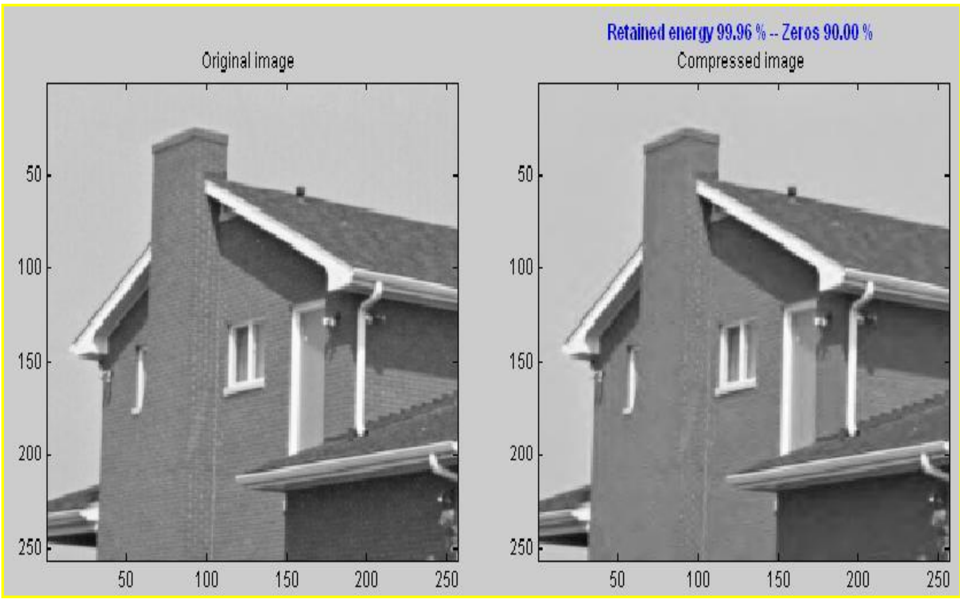


Figure 71d: Image 4 compression using D8

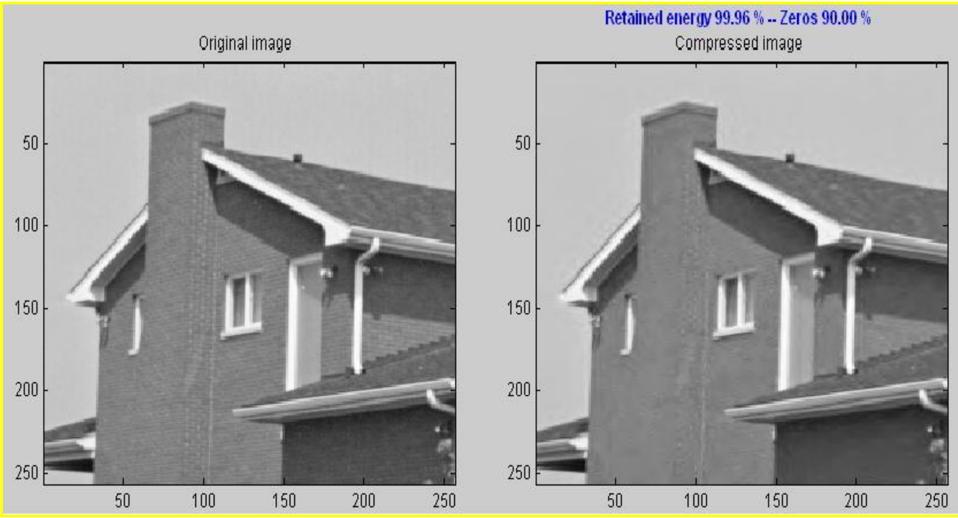


Figure 71e: Image 4 compression using D10

Figure 70 shows the wavelet decomposition at level three using 5 different wavelet filters. Figure 71 shows the original and compressed images using Haar, D4, D6, D8 and D10 respectively. The purpose of showing these figures is to give a clear observation that Haar is different with other Daubechies filters although Haar is one of the Daubechies type. For wavelet decomposition, from an original image, wavelet decomposes it into 3 parts (horizontal, vertical and diagonal). Level 3 decomposition means wavelet decomposes an image three times.

Table 31: Analysis at CR=5:1, using D8 at level 3

RE(%)	99.92	99.93	99.92	99.92	99.92	99.92	99.93
NOZ(%)	80	80	80	80	80	80	80
Threshold Horizontal (H)	4.912 17.32 81.51	4.912 17.32 81.51	4.912 17.32 81.51	5.69 20 88	4.1 15 75	5.69 20 88	4.1 15 75
Threshold Diagonal (D)	2.558 8.148 52.49	3 9 55	2.2 7.2 48	2.558 8.148 52.49	2.558 8.148 52.49	2.2 7.75 50	3.05 9.3 58
Threshold Vertical (V)	5.965 18.93 78.34	4.7 16.5 72	8.5 24 90	4.7 16.5 72	8.5 24 90	5.965 18.93 78.34	5.965 18.93 78.34
MSE	31.7394	31.7226	32.1912	30.3594	32.44	33.1485	31.1697
RMSE	5.6338	5.6323	5.6737	5.5099	5.6956	5.7575	5.583
PSNR	33.1148	33.1171	33.0534	33.3079	33.02	32.9262	33.1935

RE – Retained Energy
NOZ- Number of Zeros

For Image 4 at compression ratio 5:1, the lowest MSE observed is 30.3594 with PSNR 33.3079. The blued-column is the best result when we fix the threshold value for diagonal part. From the result, the lowest error obtained when the percentage of zeros increases (horizontal) and decreases (vertical).

Table 32: Result summary for Image 4 for CR 5:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	99.84	80	35.2656	5.9385	32.6573	-V
D4	99.81	80	59.8270	7.7348	30.3618	+H , -V
D6	99.85	80	45.5315	6.7477	31.5477	Fix value
D8	99.92	80	30.3594	5.5099	33.3079	+H , -V
D10	99.87	80	36.9685	6.0802	32.4525	Fix value

Comment:

+H = eliminate more coefficients diagonally

-V = eliminate few coefficients vertically

Fix value = fix value of threshold

From Table 32, D8r is the best filter to decompose Image 4 for a compression ration 5:1. From the comment, it can be said that we can reduce the error by increasing the percentage of zeros horizontally. We also could try reducing the percentage of zeros vertically.

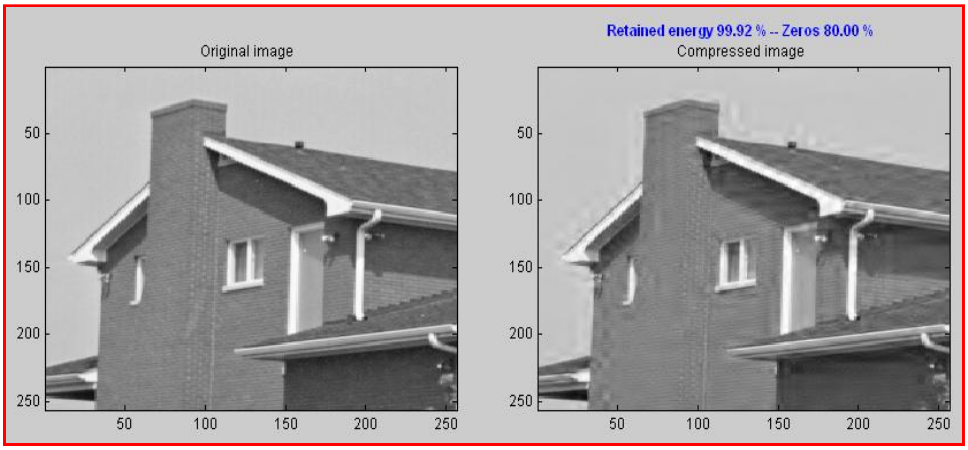


Figure 72: The best result for Image 4 compression using Haar for CR 5:1 with RMSE=5.5099 and PSNR=33.3079

Table 33: Analysis at CR=10:1, using Haar at level 3

RE(%)	99.76	99.77	99.74	99.74	99.70	99.74	99.76
NOZ(%)	90	90	90	90	90	90	90
Threshold Horizontal (H)	12.01 44.53 155.1	12.01 44.53 155.1	12.01 44.53 155.1	14 49.8 170	11.5 43 150	13.5 45 170	11.1 40 150
Threshold Diagonal (D)	4.912 14.15 73.1	5.45 17.2 77	4.5 12 68	4.912 14.15 73.1	4.912 14.15 73.1	4.5 12 68	5.2 15 78
Threshold Vertical (V)	11.89 38.74 104.5	9 36 95	14.7 43 125	9 36 95	14 40 110	11.89 38.74 104.5	11.89 38.74 104.5
MSE	48.1875	44.1250	50.5469	49.6563	48.1875	48.1875	43.3750
RMSE	6.9417	6.6427	7.1096	7.0467	6.9417	6.9417	6.586
PSNR	31.3015	31.6840	31.0939	31.1711	31.3015	31.3015	31.7584

RE – Retained Energy
NOZ- Number of Zeros

For Image 4 compression at compression ratio 10:1, now Haar produces the best result compared to other filters. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the third case where the threshold values for vertical are fixed. The lowest RMSE was observed when we reduced the percentage of zeros horizontally (-H) and increase diagonally (+D). Column 2 also has a low error due to (+D).

Table 34: Result summary for Image 4 for CR 10:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	99.76	90	43.3750	6.586	31.7584	-H , +D
D4	99.69	90	87.2793	9.3423	28.7217	-H , +D
D6	99.78	90	64.34	8.0212	30.0460	-H
D8	99.80	90	52.8417	7.2692	30.9010	Fix D
D10	99.83	90	49.1478	7.0106	31.2158	-H

Comment:

+D = eliminate more coefficients diagonally

-H = eliminate few coefficients horizontally

Fix D = fix value of threshold

From Table 34, the result agrees that for Image 4 ("House"), for a fix threshold value vertically, we could eliminate more coefficients diagonally and eliminate few coefficients horizontally in order to get the lowest error and highest PSNR.

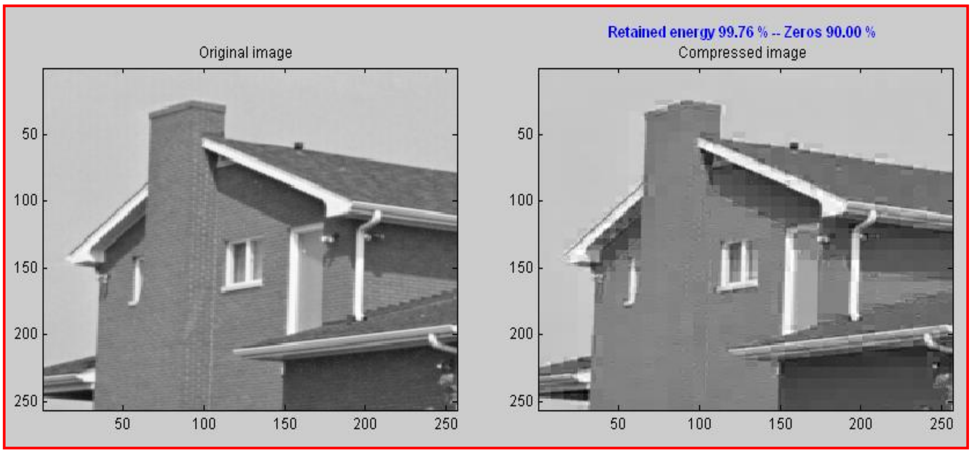


Figure 73: The best result for Image 4 compression using Haar for CR 10:1 with RMSE=6.586 and PSNR=31.7584

Table 35: Analysis at CR=20:1, using Haar at level 3

RE(%)	99.48	99.50	99.46	99.48	99.48	99.46	99.51
NOZ(%)	95	95	95	95	95	95	95
Threshold Horizontal (H)	22.46 71.01 226.3	22.46 71.01 226.3	22.46 71.01 226.3	23.6 73 235	21.5 67 215	23.4 74 250	21.5 66 215
Threshold Diagonal (D)	8.449 28.75 111.6	8.8 30 120	8 26 100	8.449 28.75 111.6	8.449 28.75 111.6	8 26 100	8.7 30 120
Threshold Vertical (V)	25.48 73.95 155.9	23.5 70 145	27 77 180	23.5 70 145	27 77 180	25.48 73.95 155.9	25.48 73.95 155.9
MSE	89.1875	73.1406	89.1875	73.5156	78.1094	89.1875	78.0937
RMSE	9.4439	8.5522	9.4439	8.5741	8.8380	9.4439	8.8371
PSNR	28.6278	29.4892	28.6278	29.4670	29.2038	28.6278	29.2046

RE – Retained Energy
NOZ- Number of Zeros

For Image 4 compression at compression ratio 20:1, now Haar produces the best result compared to other filters. From the observation, the blued-column was chosen for the best result because it has the lowest RMSE and highest PSNR. So the column belongs to the second case where the threshold values for horizontal are fixed. The lowest RMSE was observed when we reduced the percentage of zeros vertically (-V) and increase diagonally (+D). Other 2 columns also satisfied by the observation where column 4 (-V) and column 7 (+D).

Table 36: Result summary for Image 4 for CR 20:1

	RE (%)	NOZ (%)	MSE	RMSE	PSNR	Comment
Haar	95.50	95	73.1406	8.5522	29.4892	+D , -V
D4	99.41	95	127.258	11.2809	27.0840	+D , -V
D6	99.51	95	95.455	9.7701	28.3328	Fix D
D8	99.61	95	96.183	9.8073	28.2998	Fix V
D10	99.60	95	75.1292	8.6677	29.3727	-V

Comment:

+D = eliminate more coefficients diagonally

-V= eliminate few coefficients horizontally

Fix V, D = fix value of threshold

From Table 36, the result agrees that for Image 4 ("House"), for a fix threshold value horizontally, we could eliminate more coefficients diagonally and eliminate few coefficients vertically in order to get the lowest error and highest PSNR.

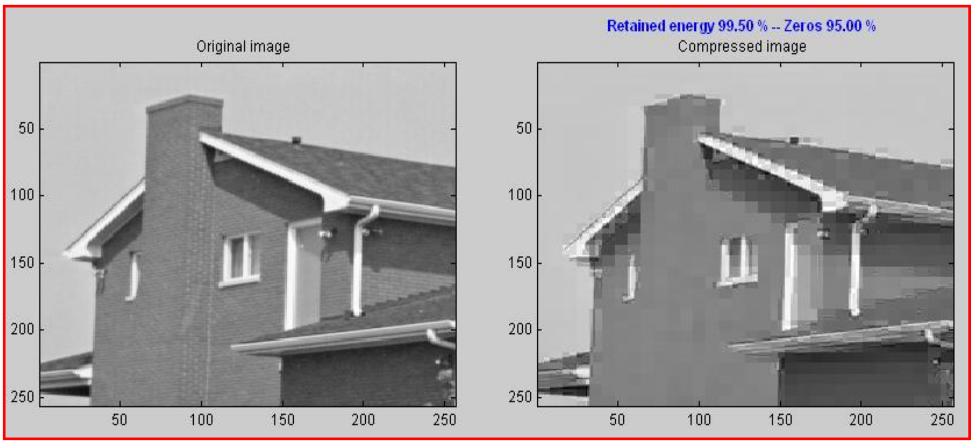


Figure 74: The best result for Image 4 compression using Haar for CR 20:1 with RMSE=8.5522 and PSNR=29.4892

Table 37: Overall Result

	CR	Best Filter	RE (%)	NOZ (%)	MSE	RMSE	PSNR	COMMENT
Lena	5:1	D8	99.88	80	23.2878	4.8257	34.4595	+D , -V
Lena	10:1	D8	99.42	90	52.9483	7.2766	30.8923	-V, +H
Lena	20:1	D8	99.16	95	74.2175	8.6150	29.4257	+H , -D
Cameraman	5:1	D8	99.93	80	23.2536	4.8222	34.4659	-
Cameraman	10:1	D8	99.70	90	50.222	7.0867	31.1219	+H , -V
Cameraman	20:1	D8	99.24	95	94.3366	9.7127	28.3840	+H , -V
Peppers	5:1	Haar	99.72	80	33.3750	5.7771	32.8966	-H , +D
Peppers	10:1	Haar	99.09	90	65.1875	8.0739	29.9892	-H , +V
Peppers	20:1	D4	98.48	95	101.3799	10.0688	28.0713	-H , +V
House	5:1	D8	99.92	80	30.3594	5.5099	33.3079	+H , -V
House	10:1	Haar	99.76	90	43.3750	6.586	31.7584	-H , +D
House	20:1	Haar	95.50	95	73.1406	8.5522	29.4892	+D , -V

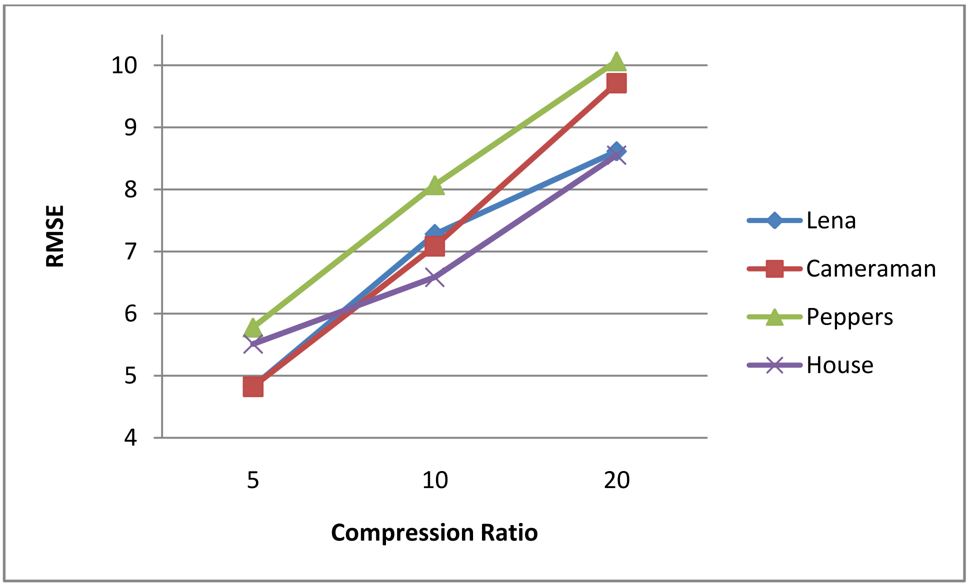


Figure 75: RMSE vs Compression Ratio for all Images

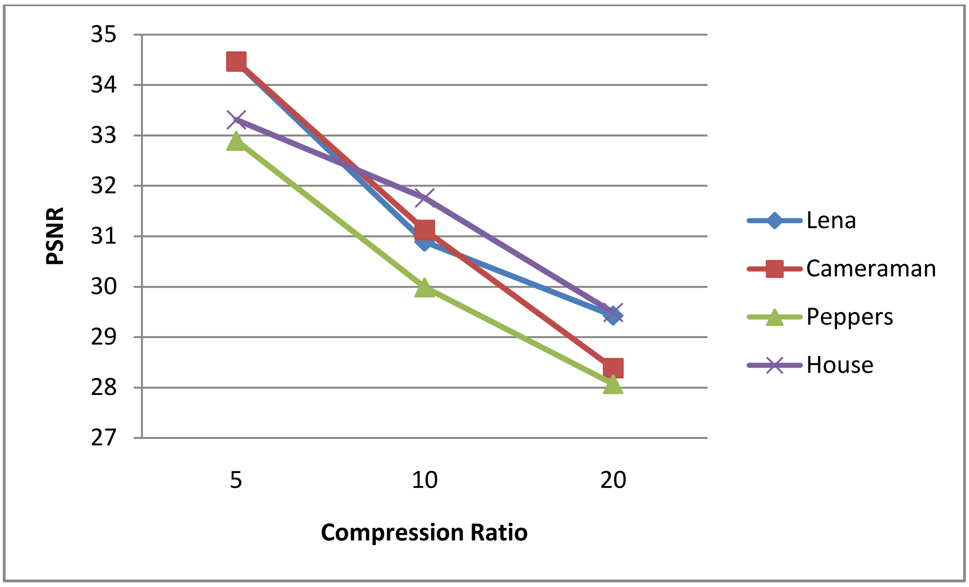


Figure 76: PSNR vs Compression Ratio for all Images

For overall result, most of the good results came from D8 filter. From Table 37 we can see for Image 1 and 2, D8 give the best results for all compression ratio values which produced the lowest error and highest PSNR. The next best filter is Haar. Haar is good for Imgae 3 and Image especially for compression ratio 10:1.

4.4 Study Case: Compress KLCI Time Series Data Using FFT

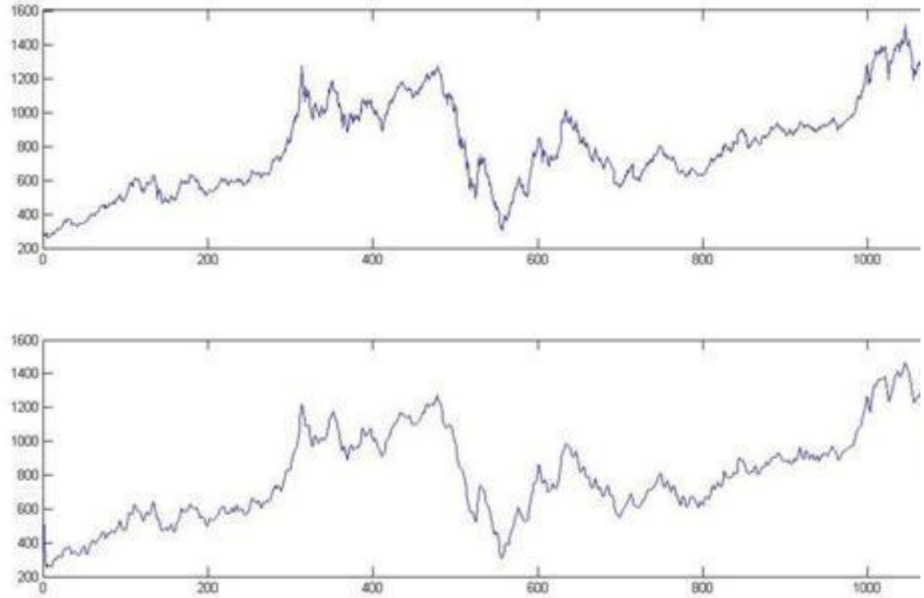


Figure 77: Compression Ratio 0.8 (Upper-Original and Lower-Compressed)

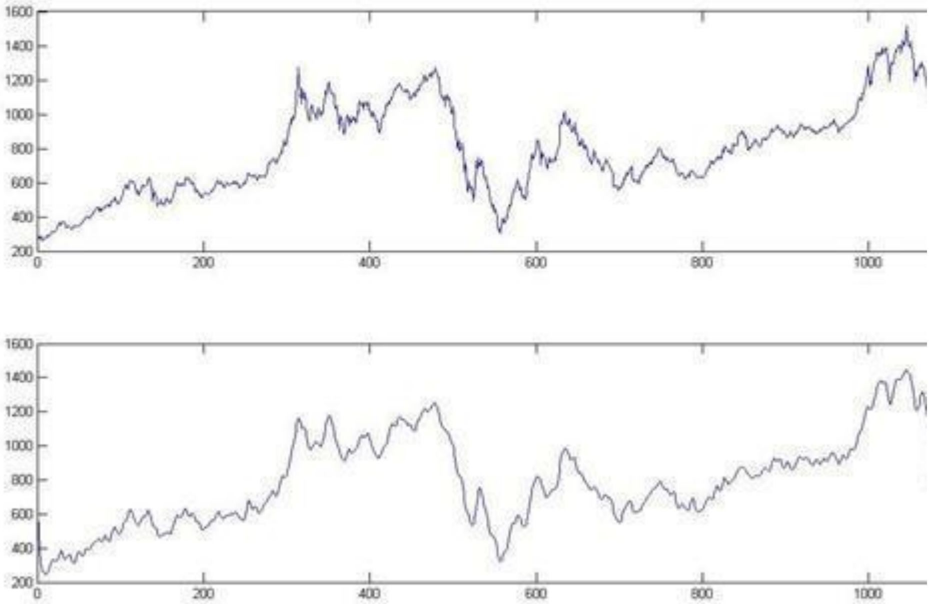


Figure 78: Compression Ratio 0.875 (Upper-Original and Lower-Compressed)

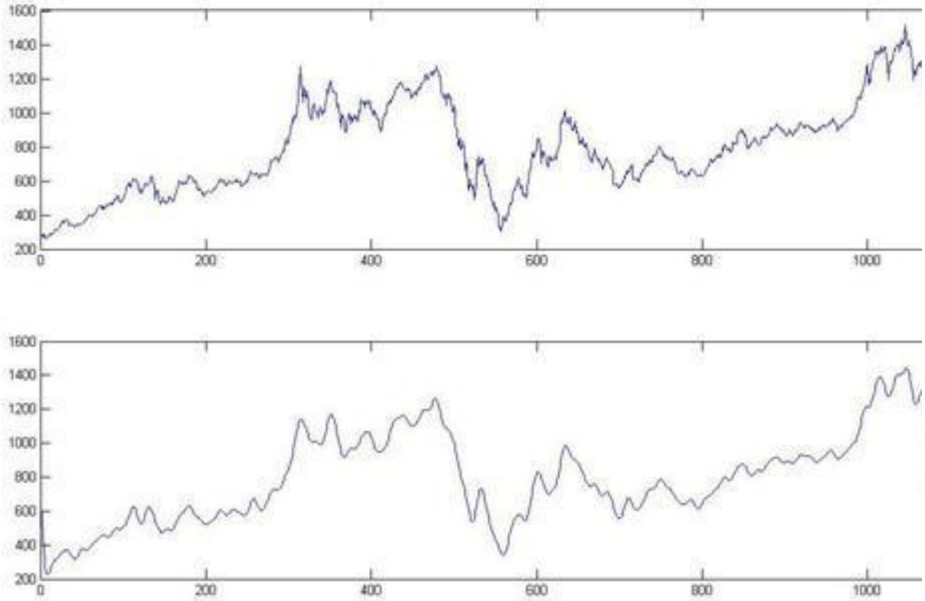


Figure 79: Compression Ratio 0.9 (Upper-Original and Lower-Compressed)

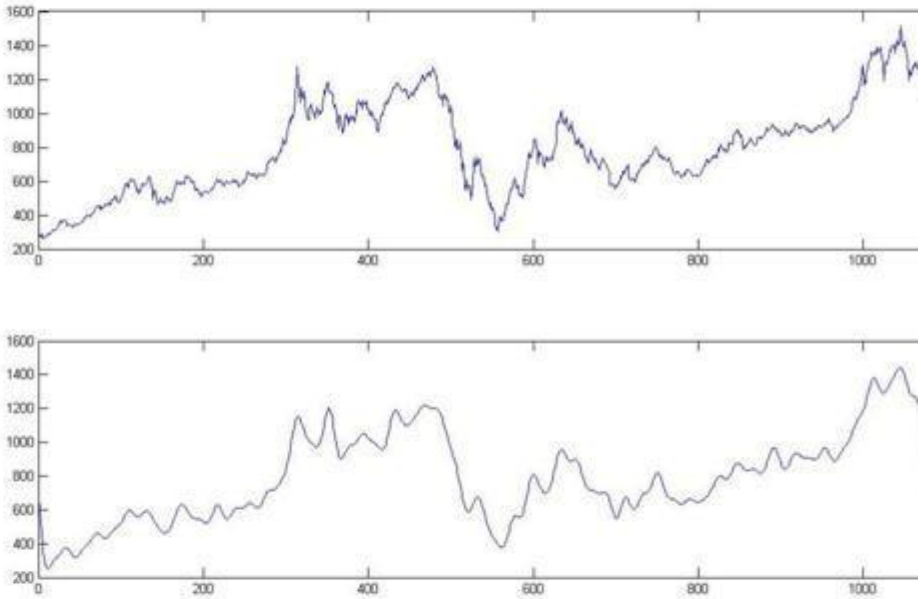


Figure 80: Compression Ratio 0.9333 (Upper-Original and Lower-Compressed)

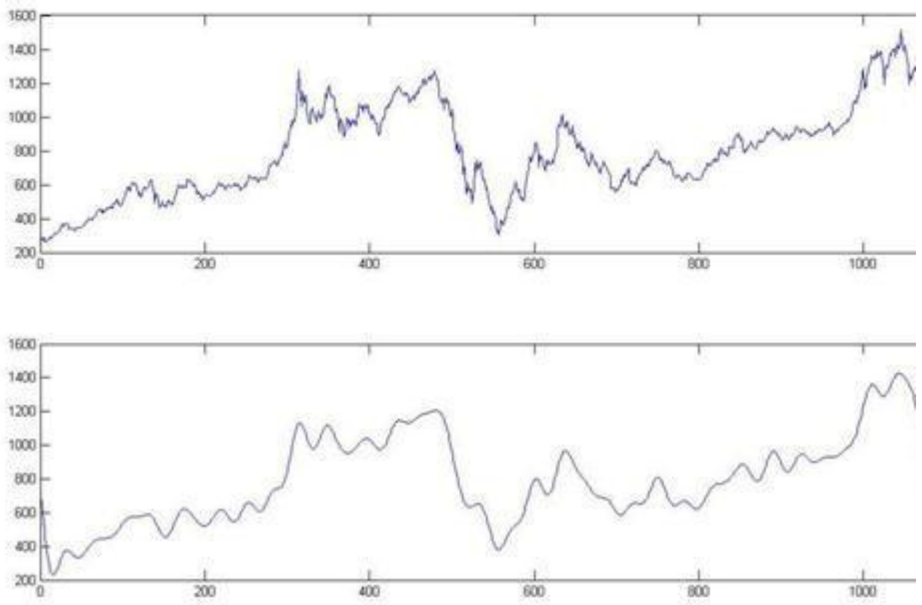


Figure 81: Compression Ratio 0.95 (Upper-Original and Lower-Compressed)

Table 38: Result Summary

CR	NOZ (%)	RMSE
0.8	80	22.3194
0.875	87.5	28.048
0.9	90	31.3602
0.9333	93.33	39.7343
0.95	95	47.5254

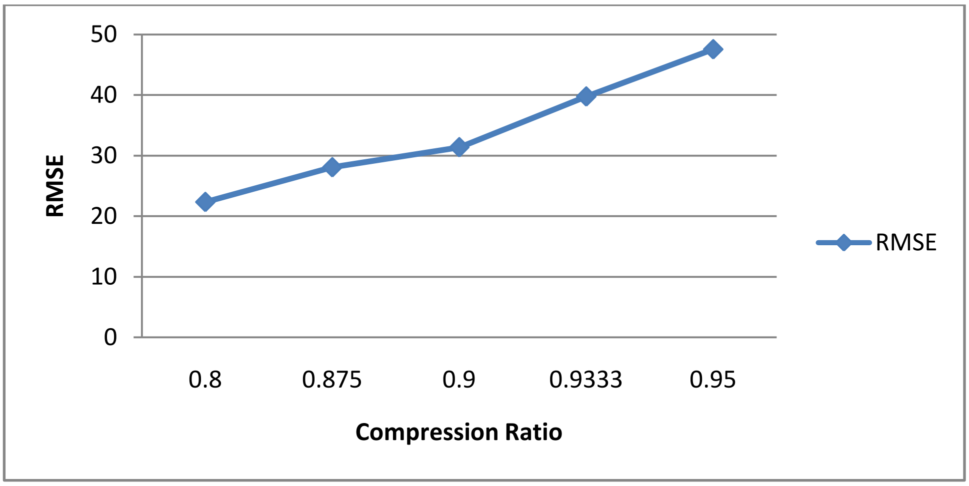


Figure 82: RMSE vs Compression Ratio

4.5 Case Study 1: Compress KLCI Time Series Data Using DWT

Table 39: Result Summary for KLCI data compression using DWT

		RMSE				
		CR 5:1	CR 8:1	CR 10:1	CR 15:1	CR 20:1
	Optimum Level	NOZ 80%	NOZ 87.5%	NOZ 90 %	NOZ 93.3%	NOZ 95%
Haar	5	14.13	20.5037	24.1762	32.7459	40.2221
D4	5	12.044	16.9450	19.5067	25.8013	32.6454
D6	6	12.0598	17.322	19.8646	25.3679	31.7075
D8	5	10.9363	15.4391	17.8469	24.5273	30.5819
D10	5	11.7917	16.8123	19.3041	25.0610	32.3237

From Table 39: D8 is found to be the best filter decompose KLCI time series data with the lowest RMSE and it is the best for all 5 different compression ratio.

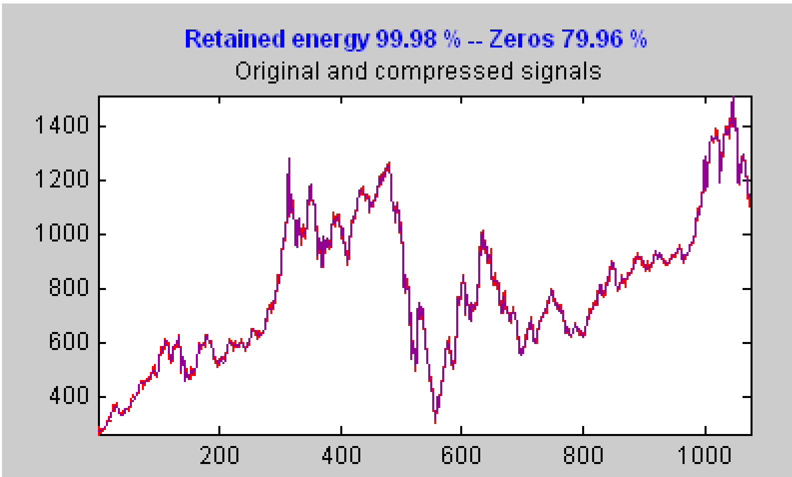


Figure 83: The best result of compression using D8 for CR 5:1

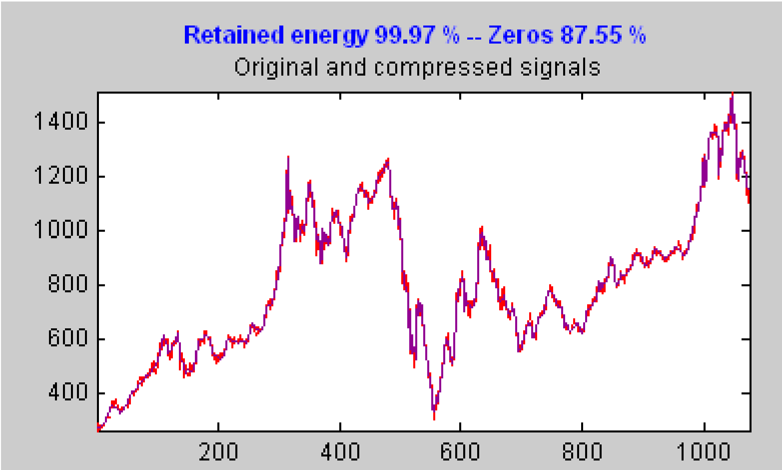


Figure 84: The best result of compression using D8 for CR 8:1

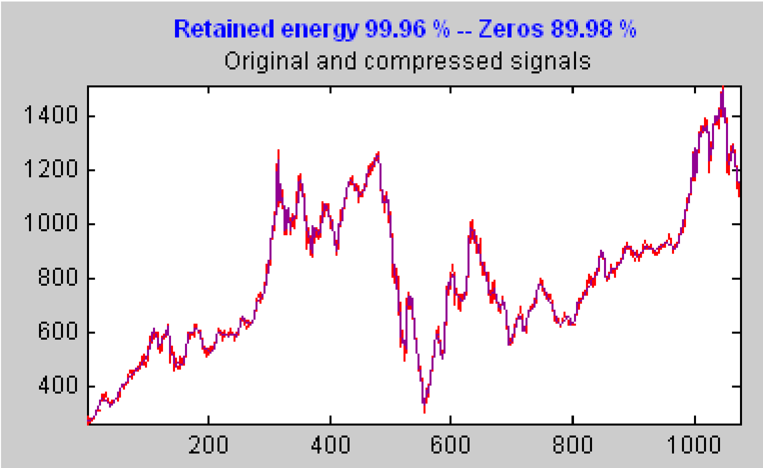


Figure 85: The best result of compression using D8 for CR 10:1

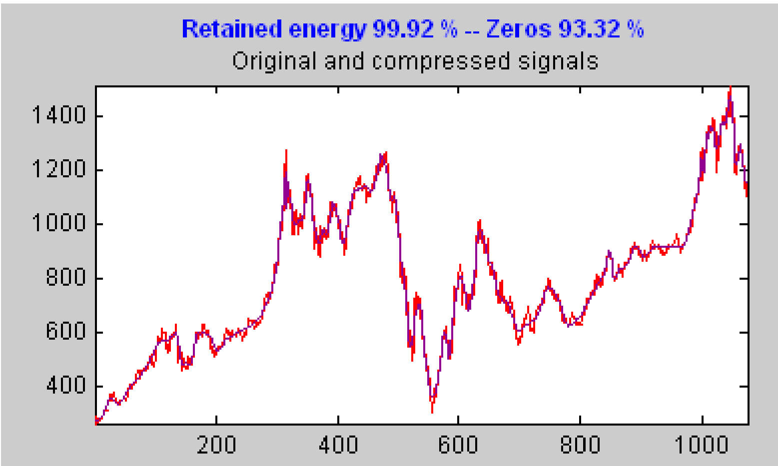


Figure 86: The best result of compression using D8 for CR 15:1

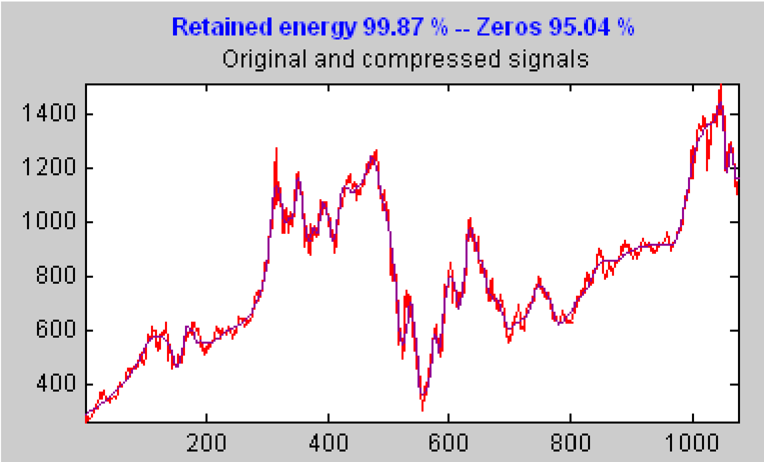


Figure 87: The best result of compression using D8 for CR 20:1

Table 40: FFT vs DWT

		RMSE				
		CR 5:1	**CR 8:1**	**CR 10:1**	**CR 15:1**	**CR 20:1**
	Optimum Level	**NOZ 80%**	**NOZ 87.5%**	**NOZ 90 %**	**NOZ 93.3%**	**NOZ 95%**
Haar	5	14.13	20.5037	24.1762	32.7459	40.2221
D4	5	12.044	16.9450	19.5067	25.8013	32.6454
D6	6	12.0598	17.322	19.8646	25.3679	31.7075
D8	5	10.9363	15.4391	17.8469	24.5273	30.5819
D10	5	11.7917	16.8123	19.3041	25.0610	32.3237
FFT	-	22.3194	28.048	31.3602	39.7343	47.5254

From Table 40, it has been proved that D8 is the best filter to decompose KLCI time series data with the lowest error. From the analysis, Daubechies filters are strongly good compared to FFT where the RMSE for FFT is more than 20 while for all Daubechies filters their RMSE is lower than 20.

4.6 Case Study 2: Electroencephalography (EEG) Characterization Using Discrete Wavelet Transform (DWT)

For this part, the actual data of EEG signal is very big and very long (continuous series of data). So, only one second of the data will be taken into account.

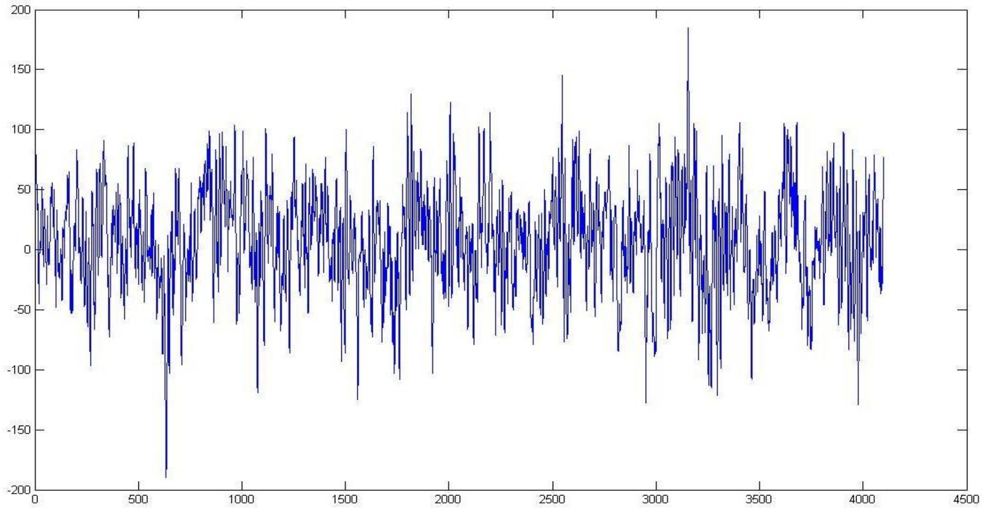


Figure 88: One second of EEG data (over 4000 coefficients)

For clearer observation, that one second of EEG data will be separated into three parts which are;

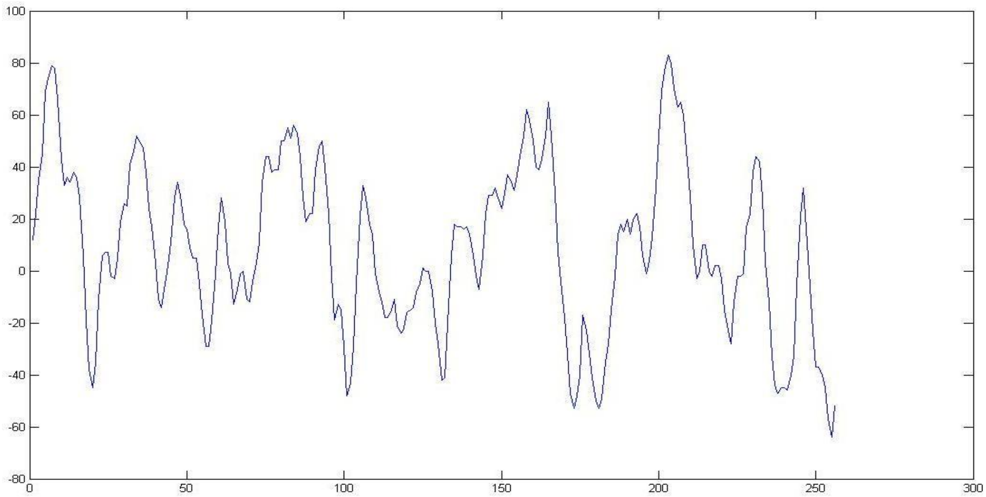


Figure 89: Sample of EEG data (256 coefficients)

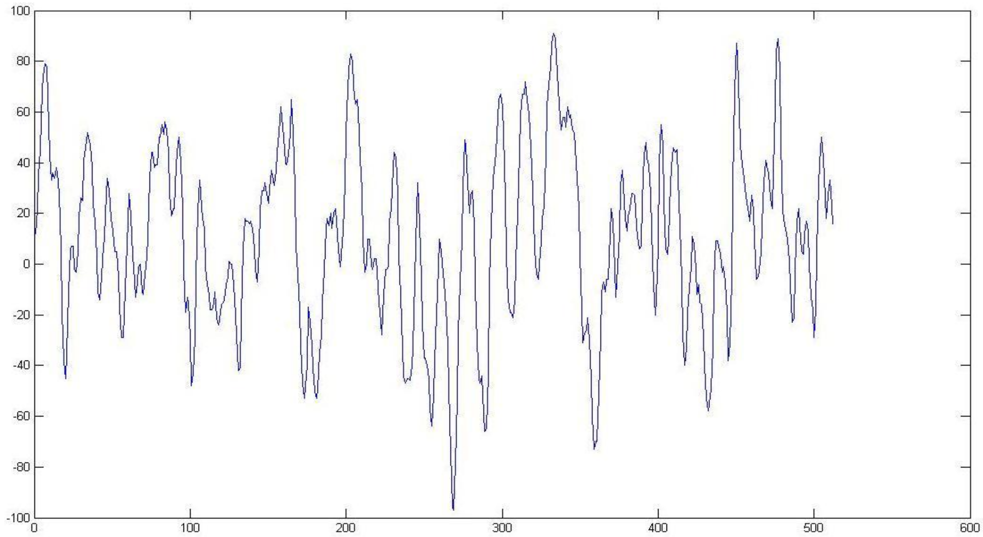


Figure 90: Sample of EEG data (512 coefficients)

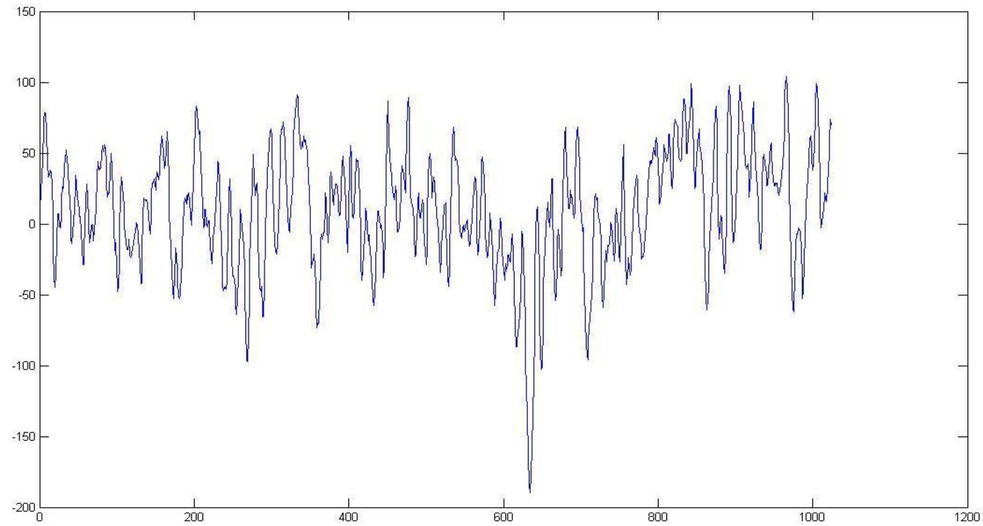


Figure 91: Sample of EEG data (1024 coefficients)

4.6.1 EEG (1024 coefficients) characterization using Haar

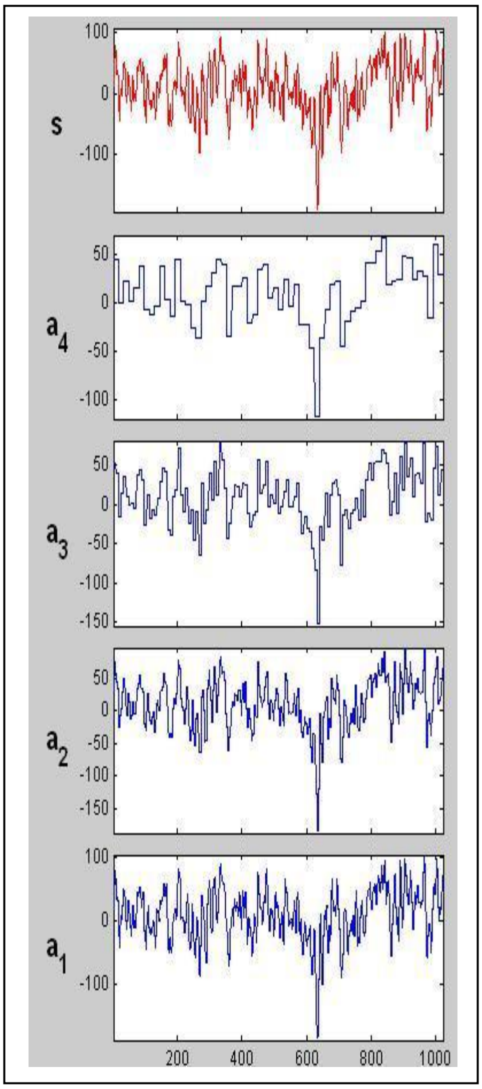


Figure 92a: Signal and Approximations by using Haar at level 4

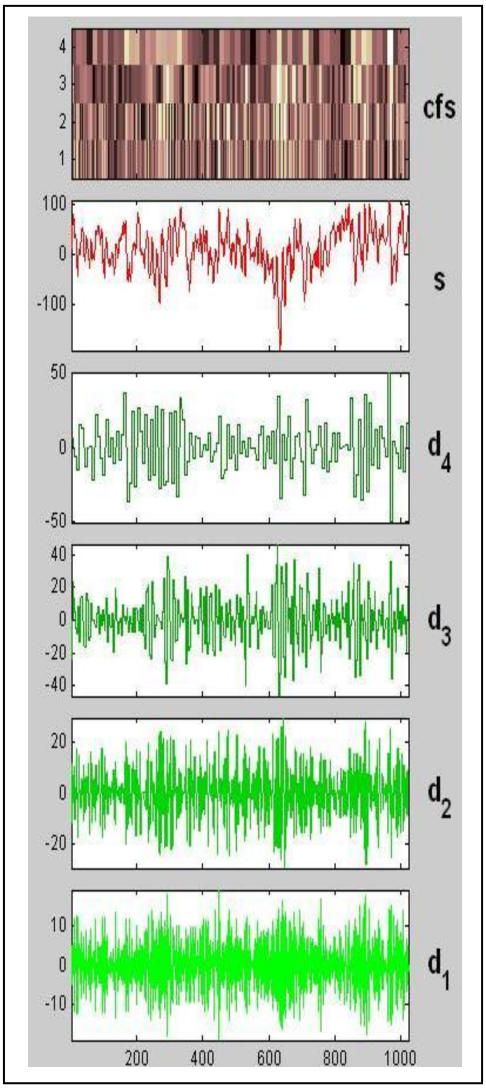


Figure 92b: Signal and Details by using Haar at level 4

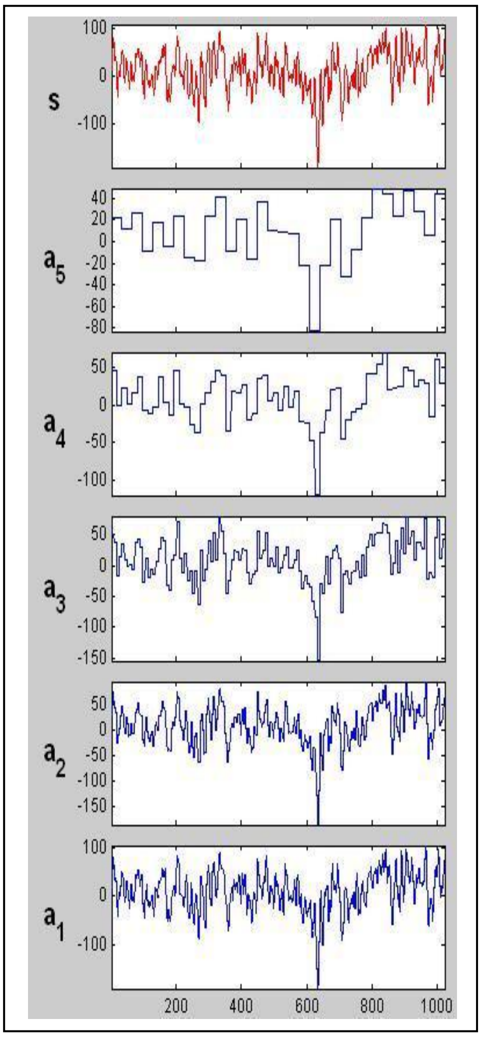


Figure 93a: Signal and Approximations by using Haar at level 5

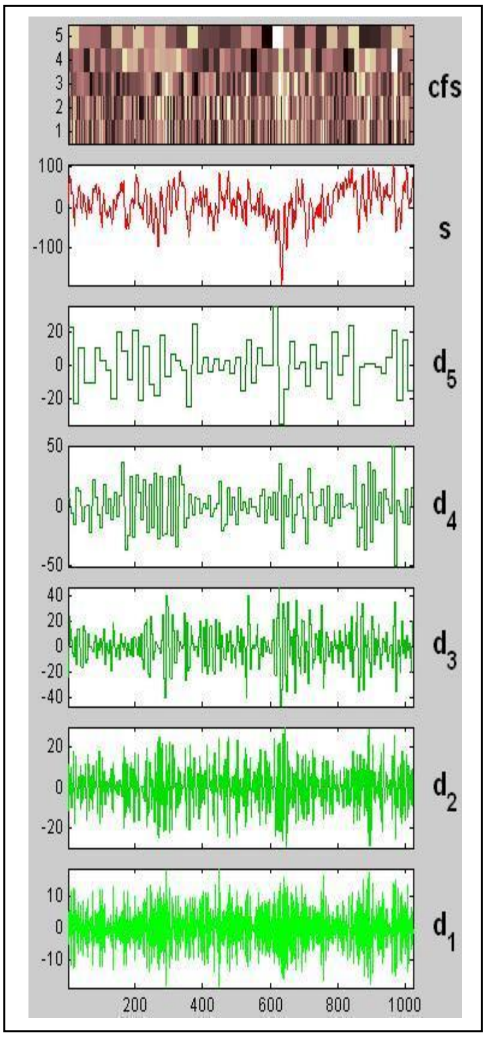


Figure 93b: Signal and Details by using Haar at level 5

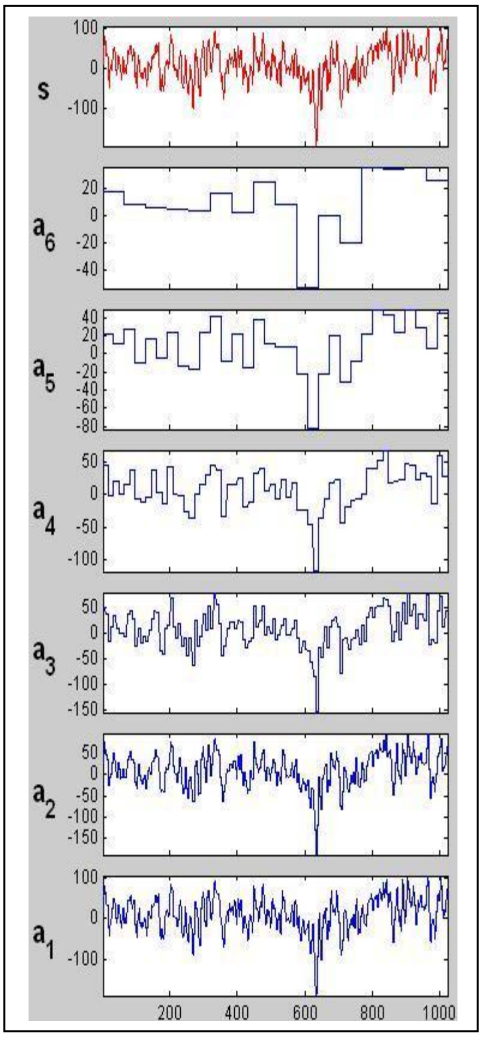


Figure 94a: Signal and Approximations by using Haar at level 6

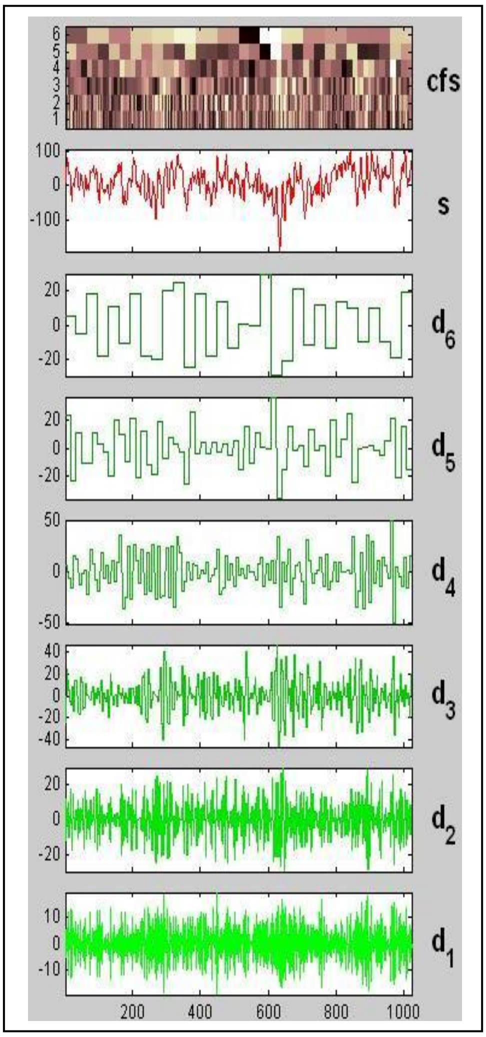


Figure 94b: Signal and Details by using Haar at level 6

4.6.2 EEG (1024 coefficients) characterization using D4

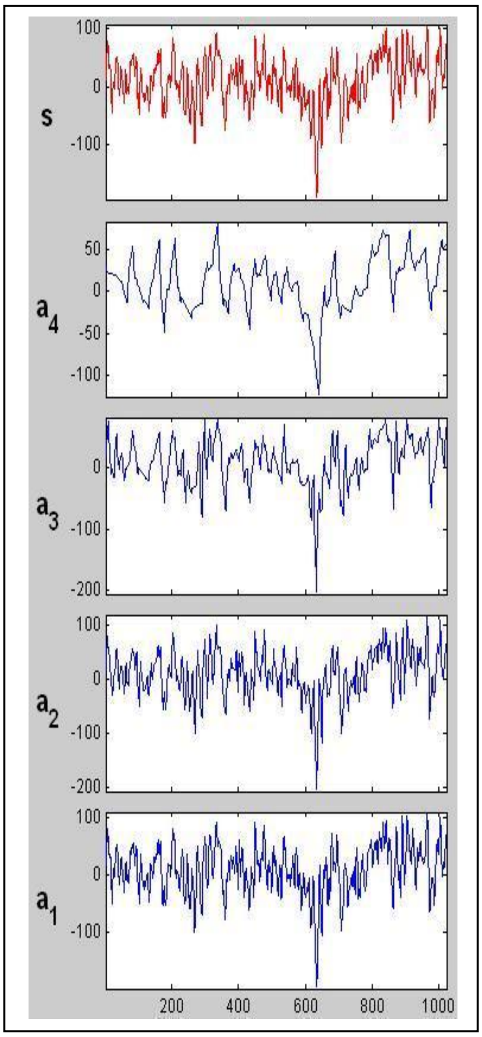


Figure 95a: Signal and Approximations by using D4 at level 4

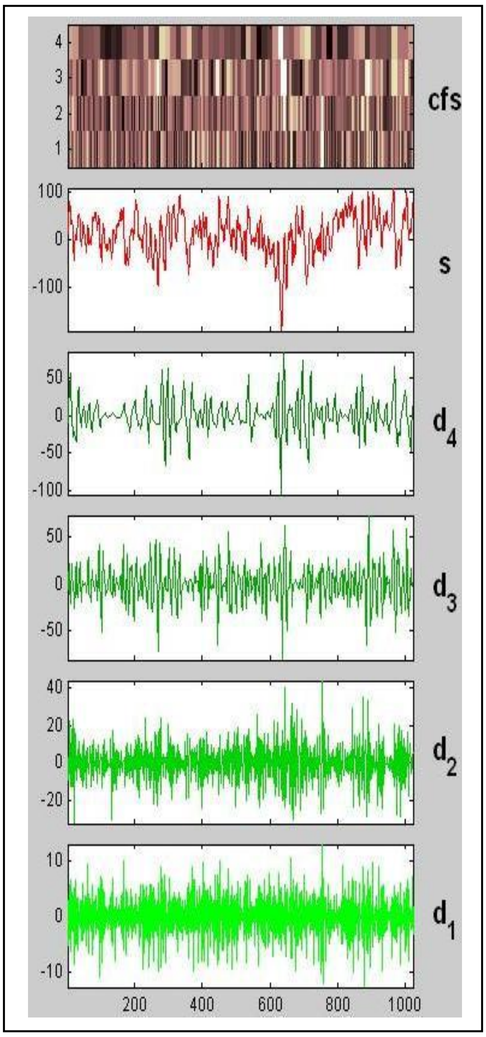


Figure 95b: Signal and Details by using D4 at level 4

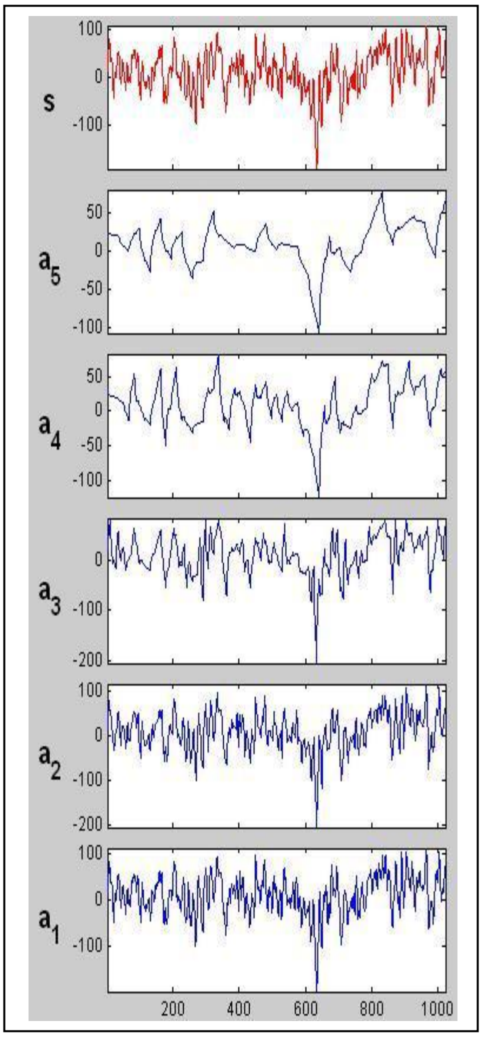


Figure 96a: Signal and Approximations by using D4 at level 5

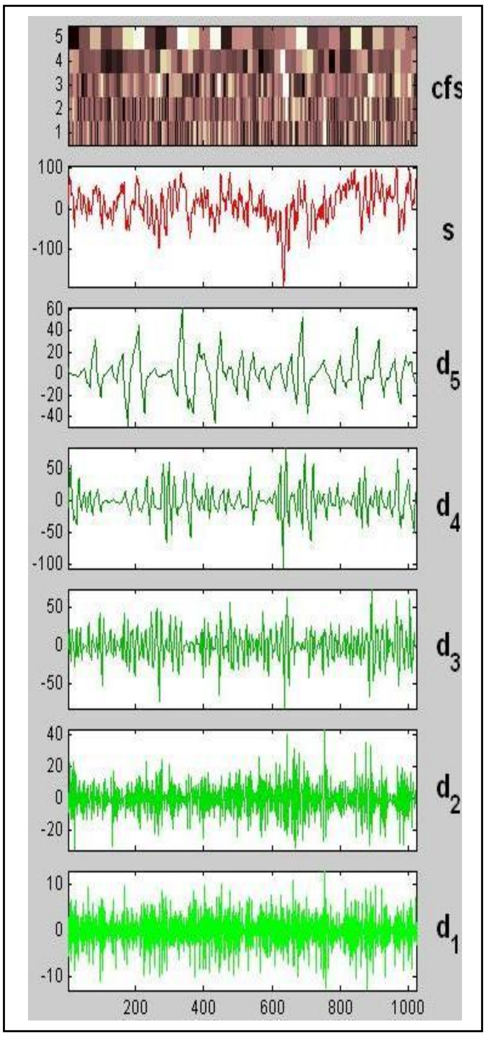
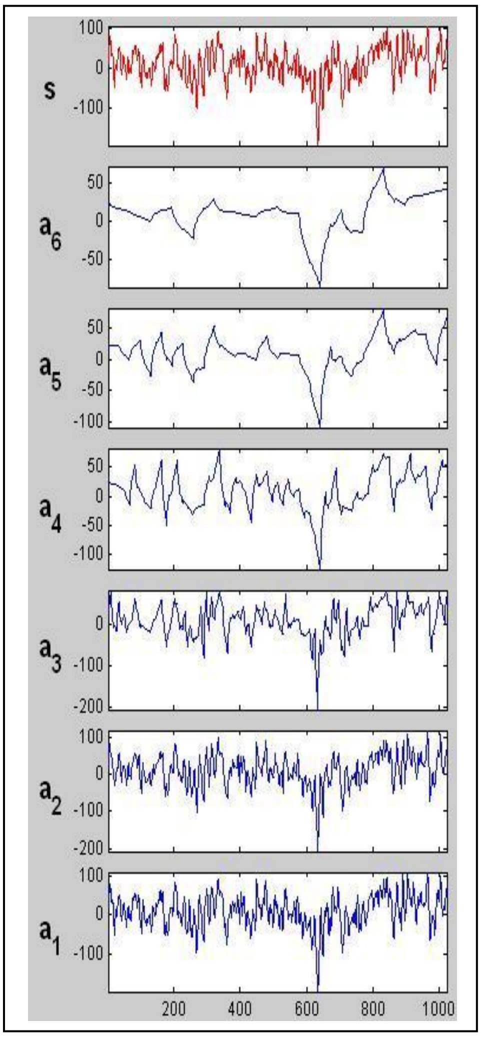


Figure 96b: Signal and Details by using D4 at level 5



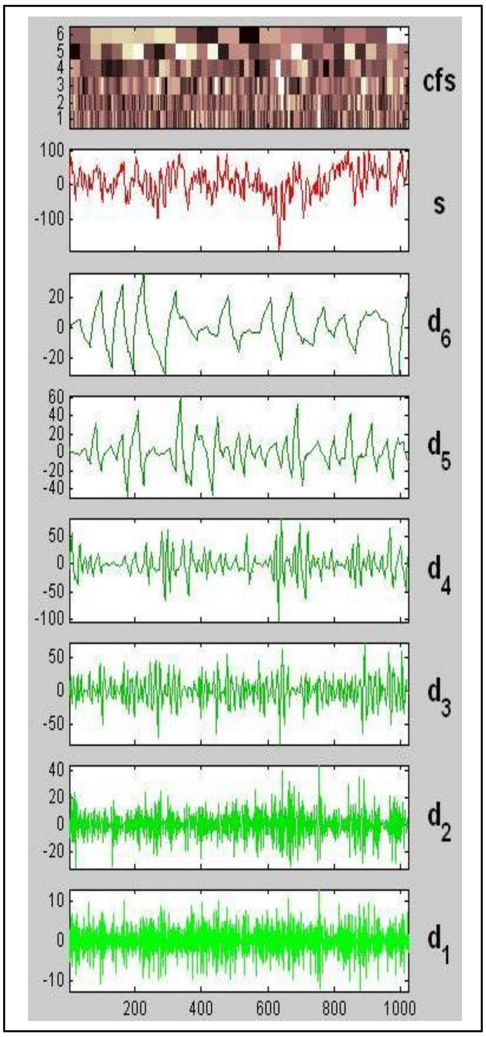


Figure 97a: Signal and Approximations by using D4 at level 6

Figure 97b: Signal and Details by using D4 at level 6

4.6.3 EEG (1024 coefficients) characterization using D6

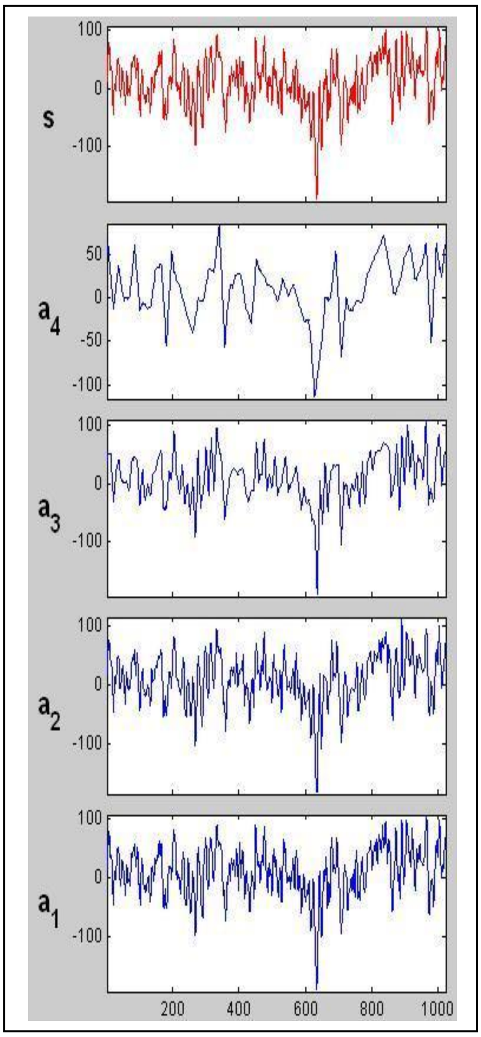


Figure 98a: Signal and Approximations by using D6 at level 4

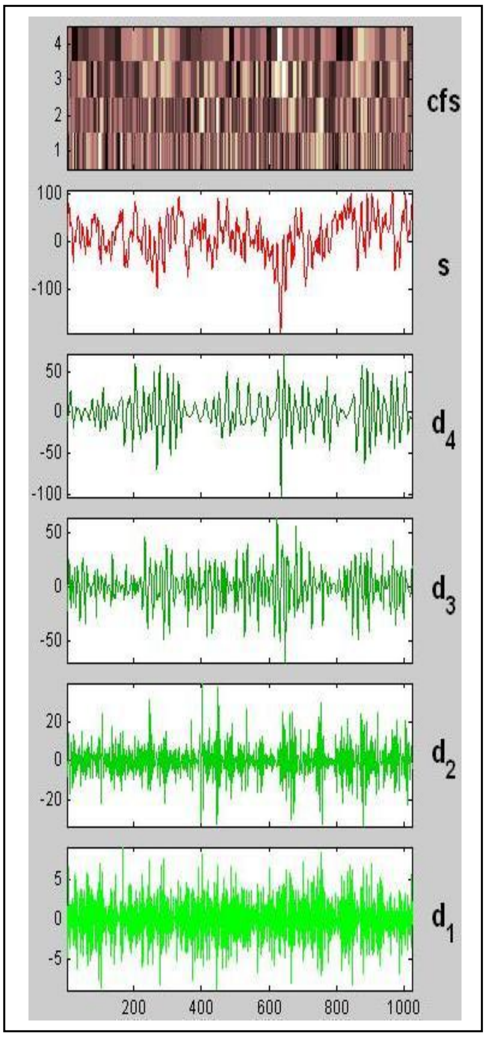


Figure 98b: Signal and Details by using D6 at level 4

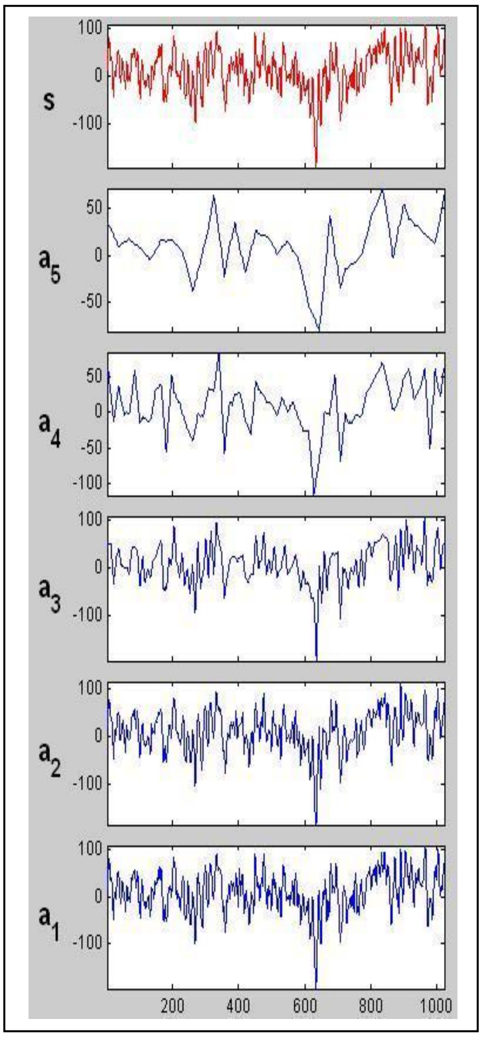


Figure 99a: Signal and Approximations by using D6 at level 5

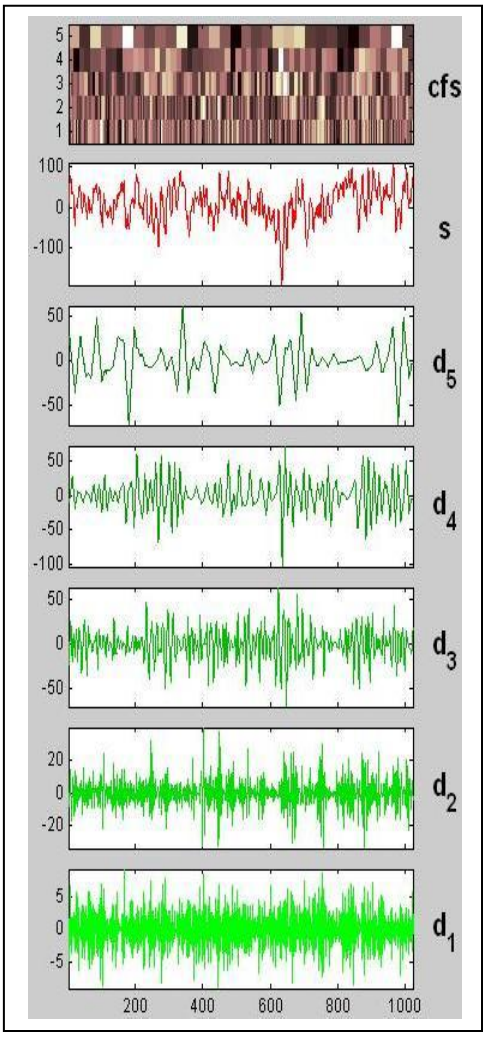
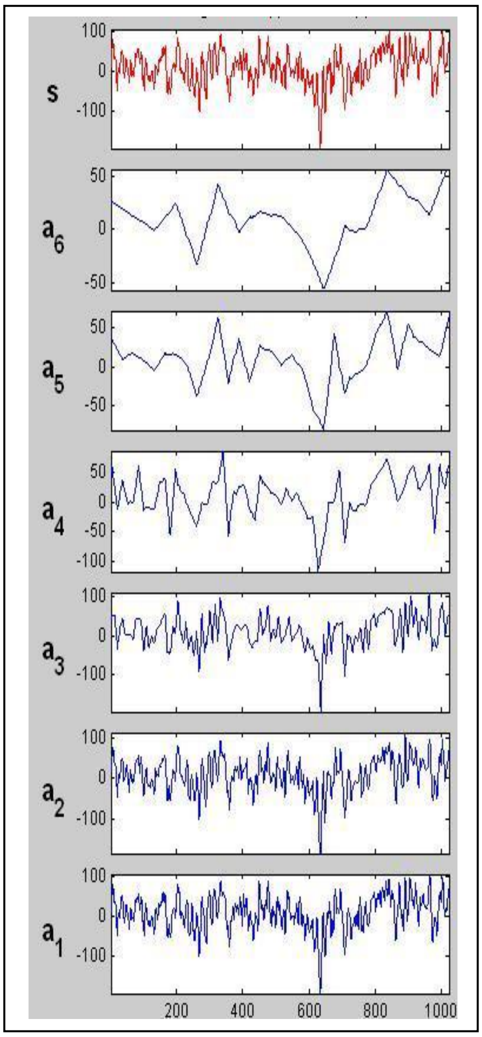


Figure 99b: Signal and Details by using D6 at level 5



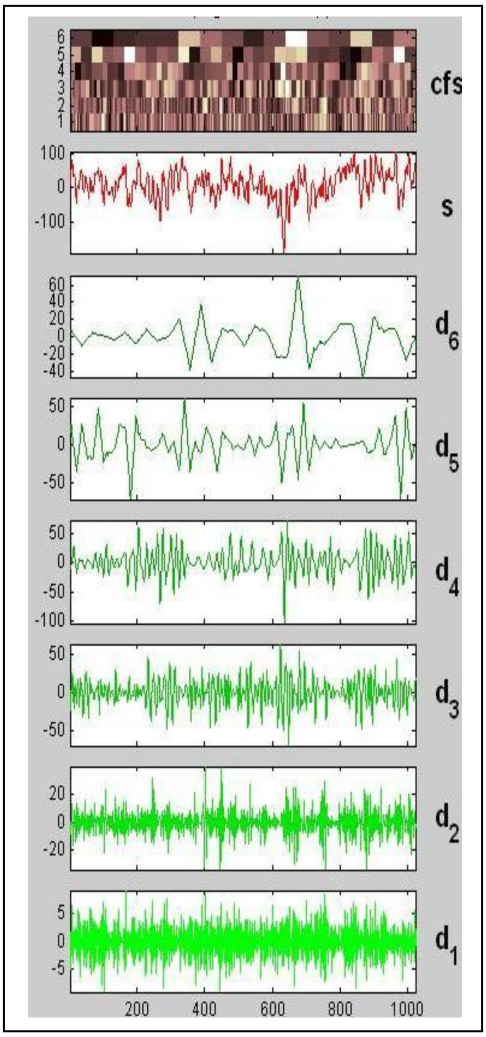
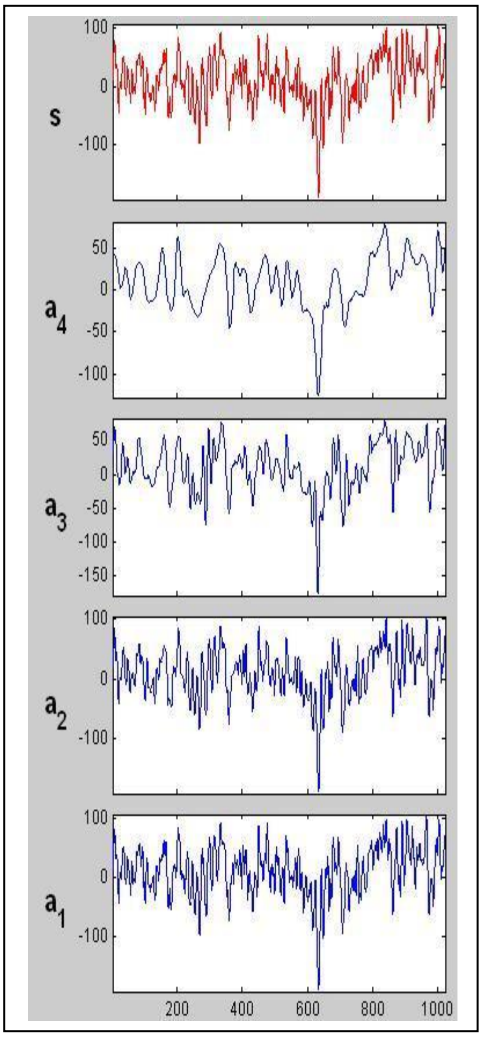


Figure 100a: Signal and Approximations by using D6 at level 6

Figure 100b: Signal and Details by using D6 at level 6

4.6.4 EEG (1024 coefficients) characterization using D8



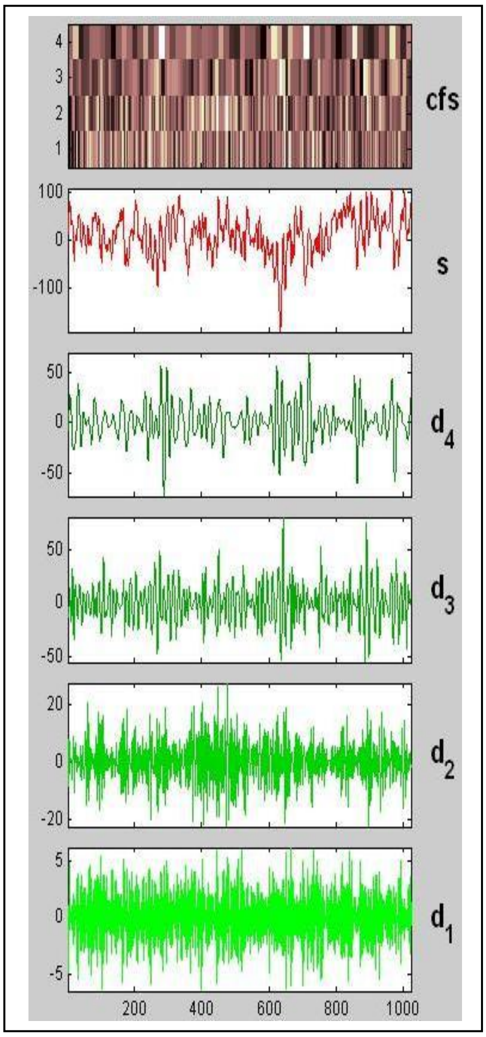


Figure 101a: Signal and Approximations by using D8 at level 4

Figure 101b: Signal and Details by using D8 at level 4

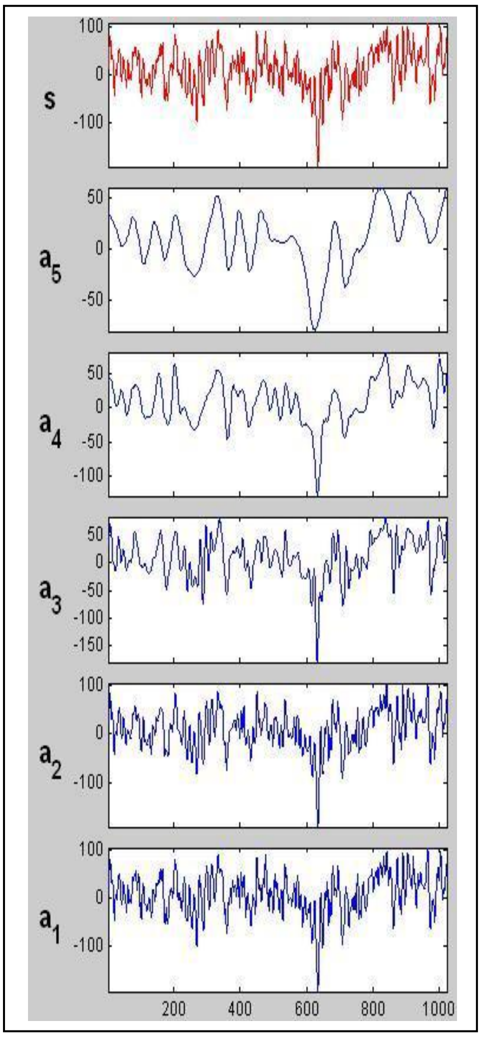


Figure 102a: Signal and Approximations by using D8 at level 5

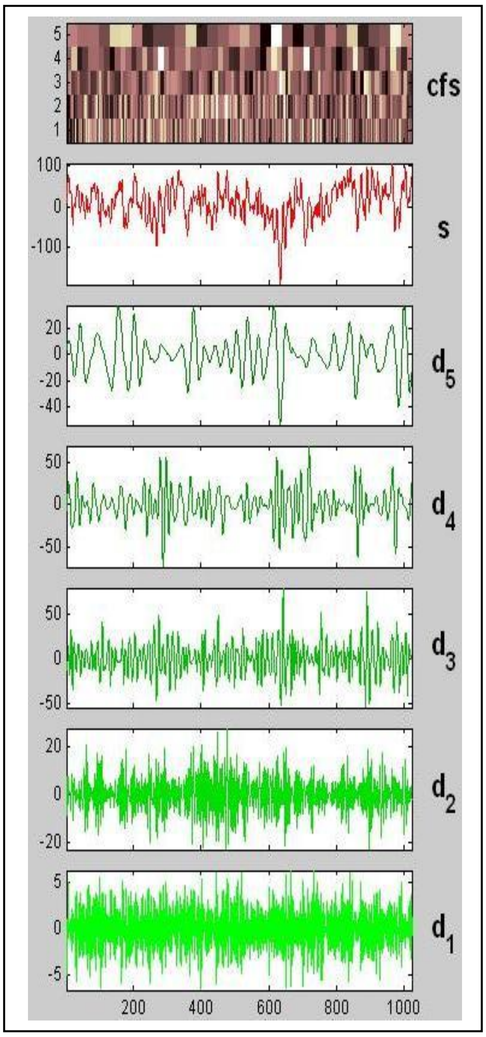
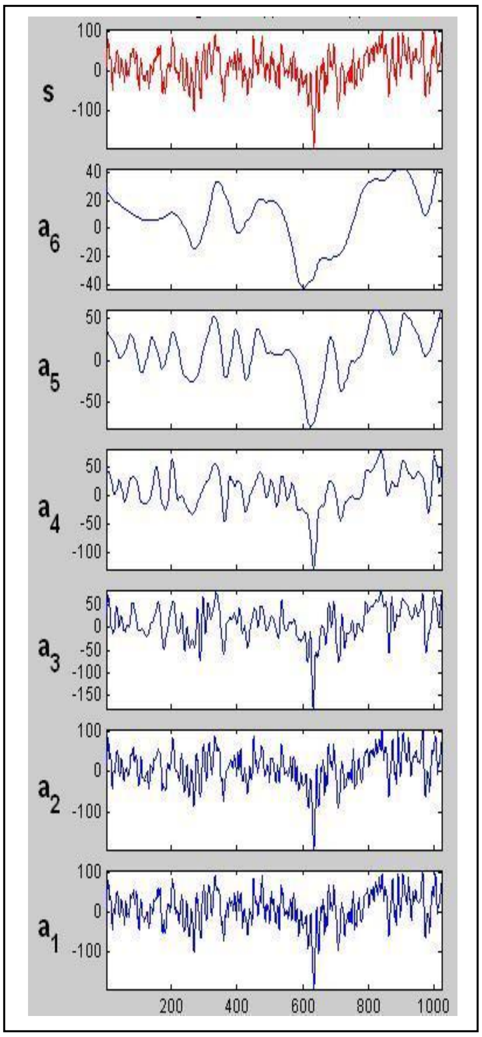


Figure 102b: Signal and Details by using D8 at level 5



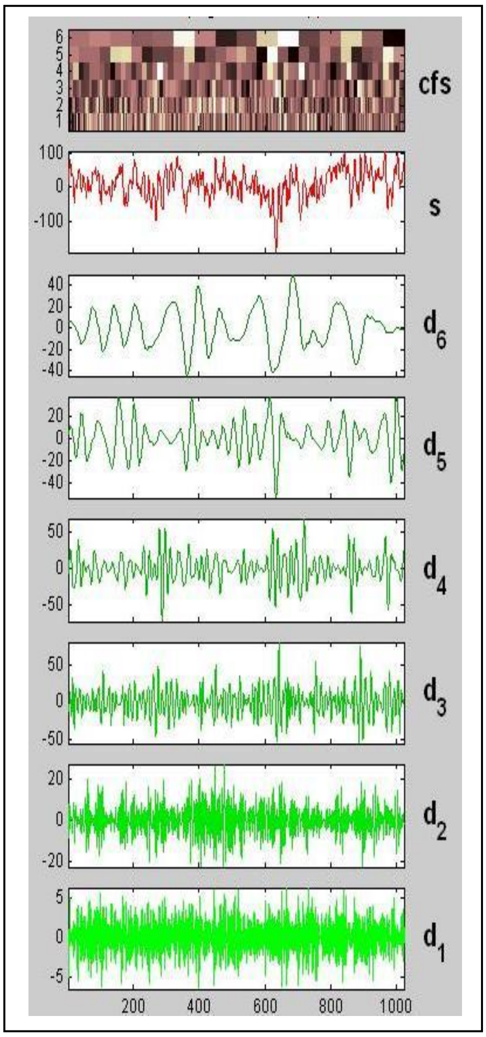


Figure 103a: Signal and Approximations by using D8 at level 6

Figure 103b: Signal and Details by using D8 at level 6

4.6.5 EEG (1024 coefficients) characterization using D10

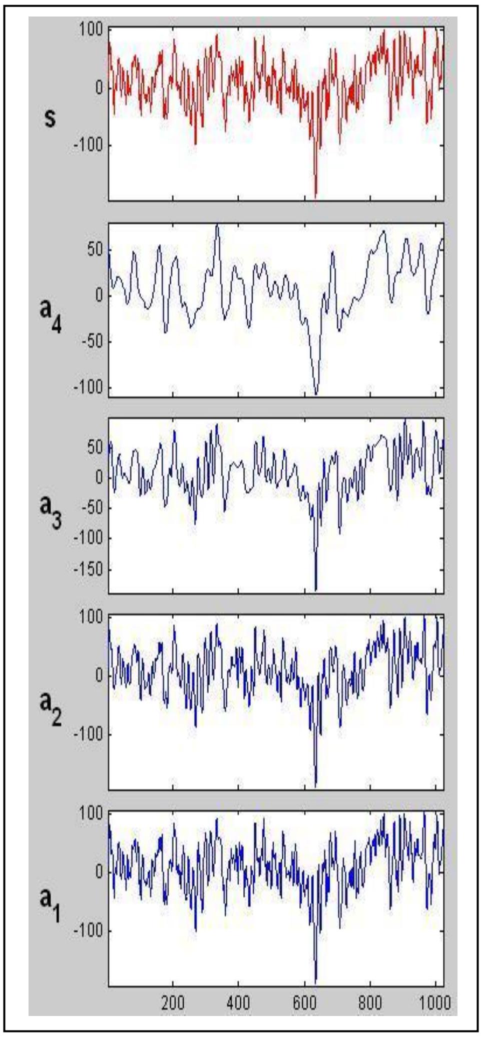


Figure 104a: Signal and Approximations by using D10 at level 4

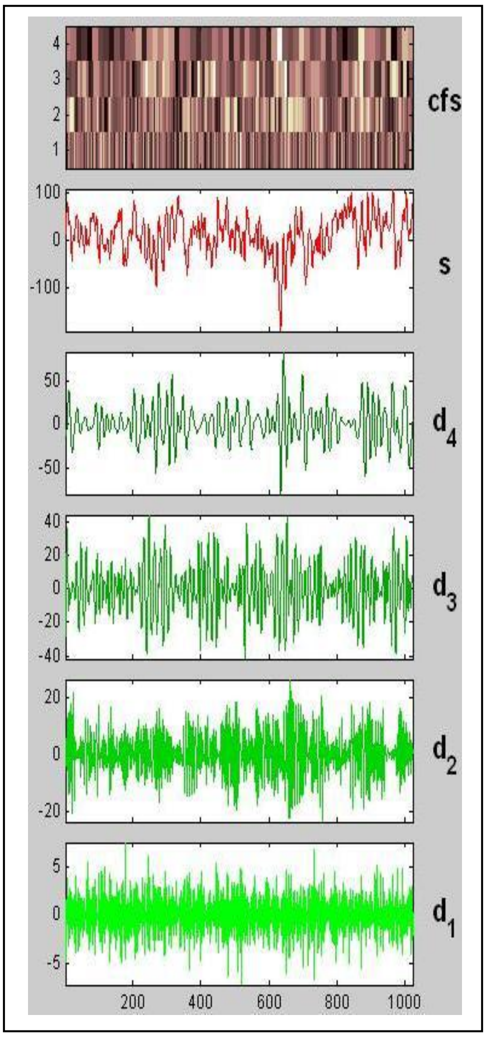


Figure 104b: Signal and Details by using D10 at level 4

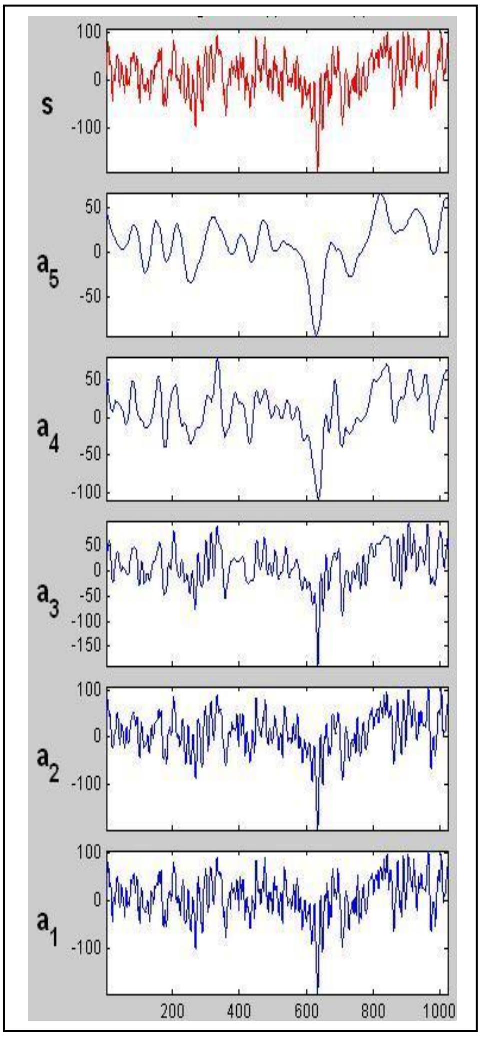


Figure 105a: Signal and Approximations by using D10 at level 5

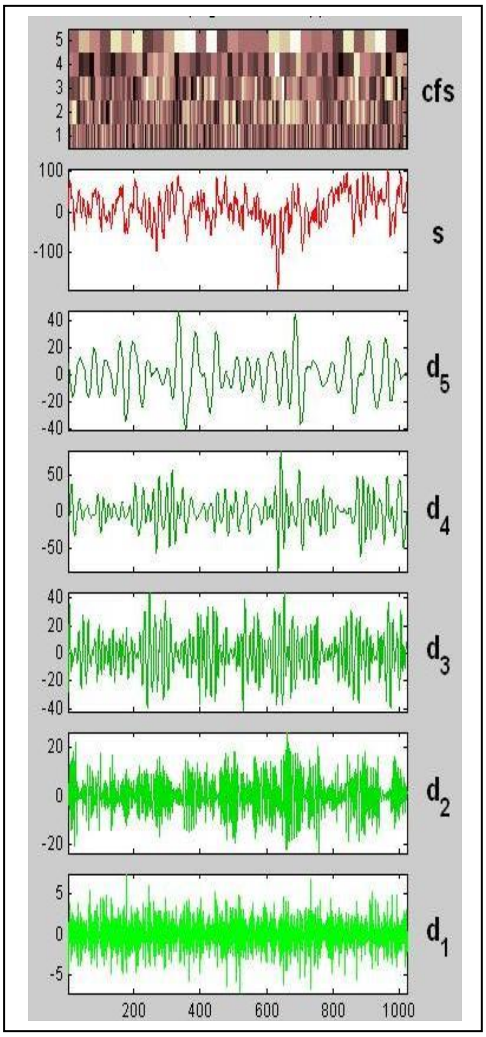
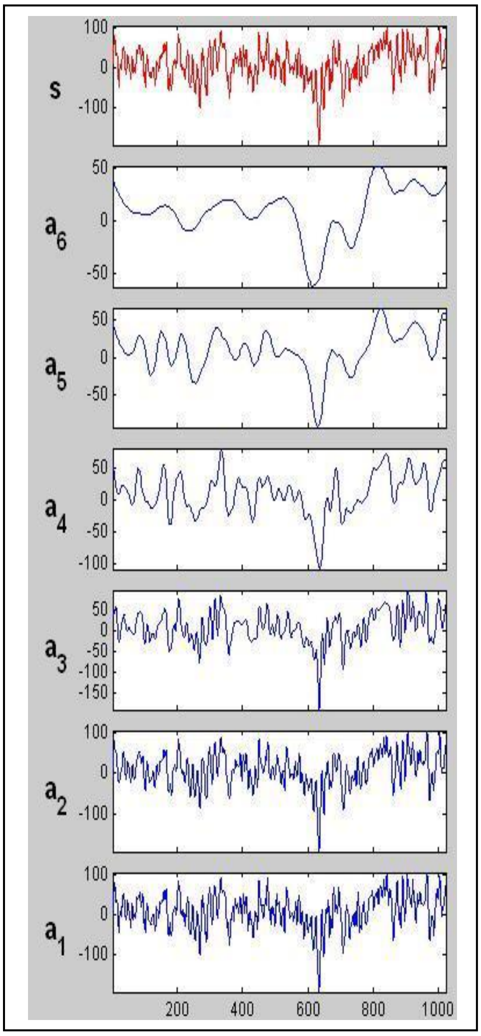


Figure 105b: Signal and Details by using D10 at level 5



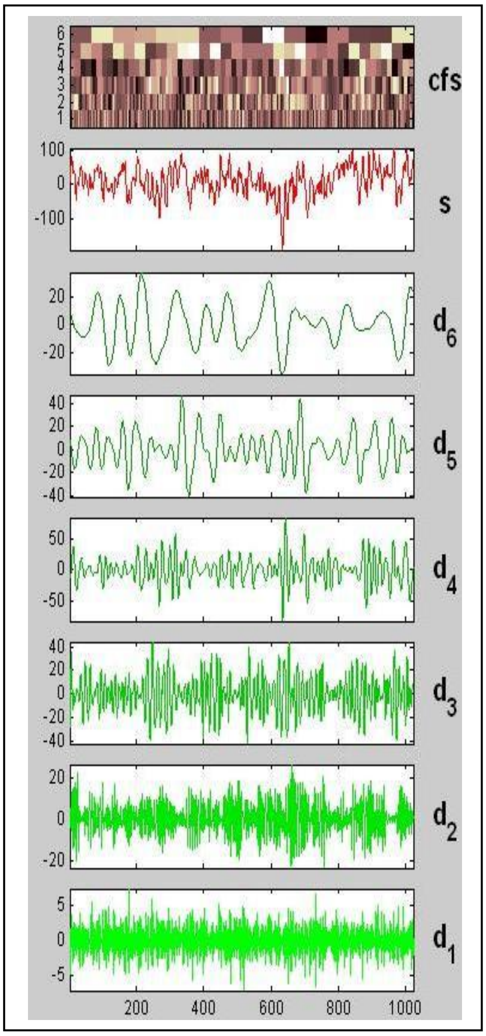


Figure 106a: Signal and Approximations by using D10 at level 6

Figure 106b: Signal and Details by using D10 at level 6

4.6.6 Classifications of EEG Waves

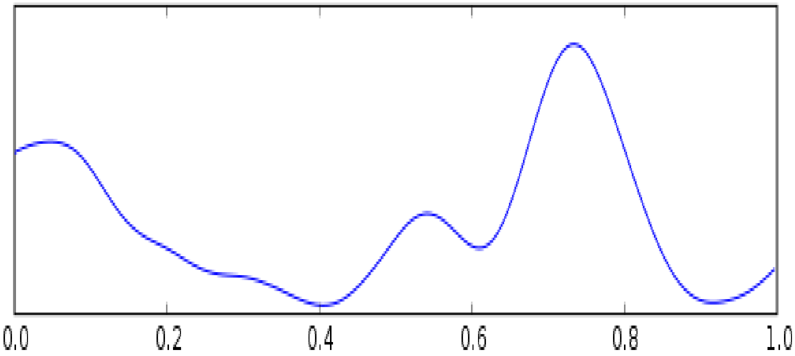


Figure 107a: Delta waves (up to 4 Hz)

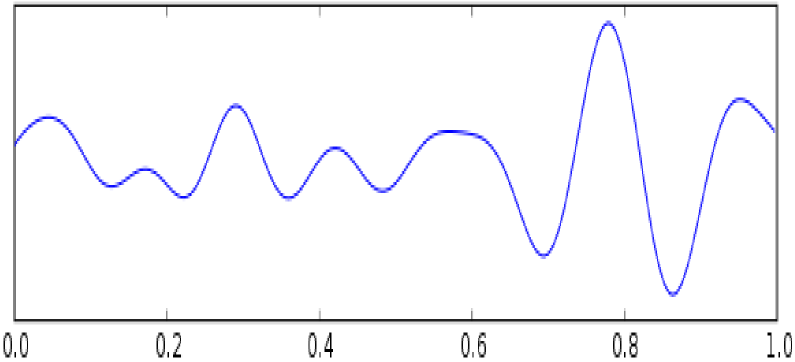


Figure 107b: Theta waves (4 – 7 Hz)

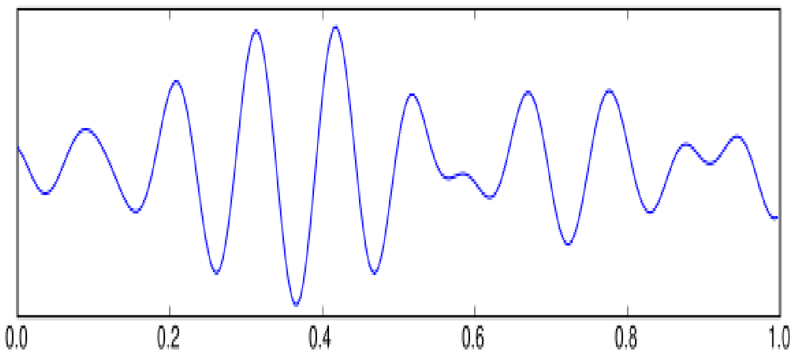


Figure 107c: Alpha waves (8 – 12 Hz)

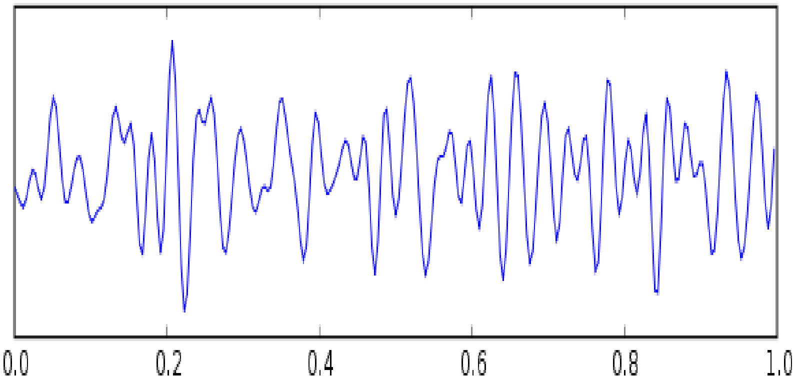


Figure 107d: Beta waves (12 – 30 Hz)

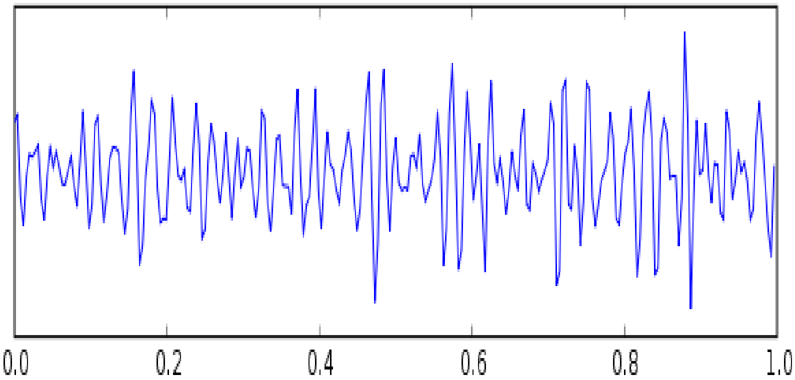


Figure 107e: Gamma waves (30 – 100+ Hz)

Table 41: Summary of Waves' Activities

Wave	Frequency (Hz)	Activities
Delta	Up to 4	-adults slow wave sleep -in babies
Theta	4-7	-young children -idling
Alpha	8-12	-relaxed/reflecting -closing the eyes
Beta	12-30	-alert/working -active,busy,anxious thinking
Gamma	30-100+	-perception that combines two different senses such as sound and sight

CHAPTER 5

CONCLUSION AND RECOMMENDATION

The main idea of data compression via fourier transform and wavelet transform is to transform a signal or data into a new format which can easily be computed, analyzed and can save space time and storage but with no accuracy degradation. In this paper, we have discussed the fourier and wavelet decomposition of 2 type of signals by using FFT, Haar and Daubechies (up to 10 filters). We used them to compress those signals to find which method gives the best result for each type of signal. Based on statistical result, Haar is the best for signals with step or block function. For sine or cosine based signal, FFT gives a quite impressive result and Daubechies give a good result for"Mishmash" type of signal. Roughly, all methods give a good result in term of MSE, RMSE and compression ratio (CR)..

For image compression, D8 is the best for most of the tested images with the lowest error and highest PSNR. For the case study which is KLCI time series data, D8 also the best filetr to decompose that signal. For future work, more filters will be used on more images to get better results and for signal compression, more type of signals will be analyzed.

REFERENCES

[1] A.M.M.A. Najih, Ramli, A.R., Prakash, V. & Syed, A.R. (2003). Speech Compression Using Discreet Wavelet Transform, NCTT 2003 Proc. 4th National Conference on Telecommunication Technology, pp. 1-4, 14-15 Jan. 2003.

[2] H. Elaydi, M. I. Jaber & M. B. Tanboura, "Speech Compression Using Wavelets," The International Arab Conference on Information Technology, pp. 1-8, 2003.

[3] J. I. Agbiny, "Discrete Wavelet Transform Techniques in Speech Processing," IEEE TENCON-Digital Signal Processing Application, pp. 514-596, 1996.

[4] J. Karam and R. Saad, "The Effect of Different Compression Schemes on Speech Signals," International Journal of Biomedical Sciences, Vol.1 No.4, pp: 230-234, 2006.

[5] O. O. Khalifa, S. H. Harding, & A. H. A. Hashim, "Audio Compression using Wavelet Transform", International Journal of Security, ISSN: 1985-2320, Computer Science Journals, Vol. 2, No. 5, pp. 17-26, September, 2008.

[6] A. Boggess and F. J. Narcowich, A First Course in Wavelets with Fourier Analysis, 2nd ed., John Wiley & Sons, Inc. Hoboken, New Jersey, 2009.

[7] C. K. Chui, An Introduction to Wavelets, Academic Press, New York, 1992.

[8] The Scientist and Engineer's Guide to Digital Signal Processing (Chapter 12), 1997. Retrieved March 2010, from http://www.dspguide.com/ch12/2.htm.

[9] J. Karam and R. Saad, “The Effect of Different Compression Schemes on Speech Signals,” International Journal of Biomedical Sciences, Vol.1 No.4, pp: 230-234, 2006.

[10] H. Elaydi, M. I. Jaber & M. B. Tanboura, “Speech Compression Using Wavelets,” The International Arab Conference on Information Technology, pp. 1-8, 2003.

[11] I. Daubechies, Ten Lectures on Wavelets, Vol 61, CBMS-NSF Reg. Con. Ser. Appl. Math.,Society for Industrial Applied Maths (SIAM), Philadelphia, PA, 1992.

[12] J. I. Agbiny, “Discrete Wavelet Transform Techniques in Speech Processing,” IEEE TENCON-Digital Signal Processing Application, pp. 514-596, 1996.

[13] Wavelet vs. Fourier Transform, 2004. Amara.com. Last Modified May 12, 2004 from http://www.amara.com/IEEEwave/IW_wave_vs_four.html

[14] M. H. Kamarudin, “On the use of Daubechies Wavelets for Signal and Image Compression: Some Numerical Evaluation,” Bachelor Thesis (Hons), Universiti Teknologi PETRONAS. In preparation, 2010a.

[15] S. A. A. Karim, M. T. Ismail, B. A. Karim, M. K. Hassan, and J. Sulaiman, “Compression KLCI Time Series Data Using Wavelet Transform,” World Engineering Congress 2010, 2nd – 5th August 2010, Kuching, Sarawak, Malaysia Conference on Engineering and Technology Education. In CD, 2010.

[16] S. A. A. Karim, B. A. Ismail, M. T. Hasan, and J. Sulaiman, “Applications of Wavelet Method in Stock Exchange Problem,” Proceedings of International Conference on Fundamental and Applied Sciences (ICFAS 2010), 15-17 June 2010, Kuala Lumpur Convention Center, 2010. In CD, 2010.

[17] S. A. A. Karim, and M. H. Kamarudin, “Signal Compression using Fast Fourier Transform (FFT) and Discrete Wavelet Transform (DWT): Some Numerical Comparison,” Preprint, 2010b.

[18] Y. Meyer, Wavelet and Operators, Cambridge University, Cambridge, 1992.

[19] J. Akhtar and M.Y.Javed, “Image Compression with Different Types of Wavelets” IEEE-ICET 2006, International Conference on Emerging Technologies held on 13-14 November 2006.

[20] Haar Wavelet, 2010. Wikipedia.com. Last Modified March, 31, 2010, from http://en.wikipedia.org/wiki/Haar_wavelet

[21] R.Merry Wavelet Theory and Application, A Literature Study. Eindhoven University of Technology, Department of Mechanical Engineering, DCT nr.2005.53, June 7,2005